

Biblioteka Główna i OINT
Politechniki Wrocławskiej



100030492

ELEMENTY INFORMATYKI

Zbigniew Huzar

HOLDING EDUKACYJNY — PAWEŁ PIETRZYK Sp. z o.o.

WYŻSZA SZKOŁA
ZARZĄDZANIA I BANKOWOŚCI

— W P O Z N A N I U —
FILIA WE WROCŁAWIU

Działająca na podstawie decyzji MEN z dnia 26.06.1992 r.
Nr DNS 3-TBM-48/92 wraz z późniejszymi zmianami



ELEMENTY INFORMATYKI

ZBIGNIEW HUZAR

ELEMENTY INFORMATYKI

Poznań-Wrocław 2004

Publikacja przeznaczona dla studentów uczelni wyższych

Wydano na podstawie

Planu Wydawniczego na rok 2004

Wyższej Szkoły Zarządzania i Bankowości z siedzibą w Poznaniu

oraz za zgodą Redaktora Naczelnego

Wydawnictwa FORUM NAUKOWE

© Copyright by PASSAT — Paweł Pietrzyk

Poznań 2004

Wydanie I

RECENZENT

Prof. zw. dr hab. Andrzej Gospodarowicz

REDAKTOR

Adriana Staniszevska

KOREKTA

Adriana Staniszevska

PROJEKT OKŁADKI

Paweł Pietrzyk

Daria Tarara



3223456/1

Wszelkie prawa zastrzeżone

Kopiowanie i powielanie w jakiegokolwiek formie

wymaga pisemnej zgody Wydawcy

WYDAWNICTWO FORUM NAUKOWE

(kontynuacja WYDAWNICTWA TERRA)

ul. Robocza 4

61-538 Poznań

www.wsizb.poznan.pl

WYDAWCA

PASSAT — Paweł Pietrzyk

ul. Kotlarska 5/1

50-150 Wrocław

ISBN 83-88544-08-X

DRUK I OPRAWA

ABEDIK S.A.

5319/G1/05

2005/0920/K

SPIS TREŚCI

PRZEDMOWA	7
I. CZYM JEST INFORMATYKA?	11
Informacja	11
Dane	14
Kodowanie danych	16
System informacyjny i system informatyczny	19
II. ALGORYTM, KOMPUTER, JĘZYK PROGRAMOWANIA	23
Algorytm	23
Maszyna Turinga	27
Człowiek czy komputer?	29
Komputer von Neumanna	32
O językach programowania	40
Translatory	46
III. PODSTAWOWE INFORMACJE O KOMPUTERZE OSOBISTYM	49
Rodzaje komputerów	49
Elementy konfiguracji sprzętowej	51
Elementy oprogramowania	61
IV. SYSTEM PLIKÓW	65
Czym są pliki?	65
Pliki tekstowe i tekstowo-graficzne	66
Pliki graficzne	67
Pliki audio (dźwiękowe)	69
Pliki video (filmowe)	71
Pliki hipertekstowe i hipermedialne	72
Pliki skompresowane	73
Pliki w systemie <i>Windows</i>	73
Kartoteki plików	75

V. INTERFEJS KOMUNIKACYJNY	79
Wprowadzenie	79
Interfejs okienkowy	81
Praca z komputerem	85
VI. SIECI KOMPUTEROWE	93
Co to jest sieć komputerowa?	93
Identyfikacja komputerów	99
Sieci otwarte	101
Światowa pajęczyna WWW	102
Usługi poczty elektronicznej	106
Inne usługi	108
VII. PODSTAWOWE ZASADY PRACY W SIECI KOMPUTEROWEJ	111
Bezpieczeństwo systemów komputerowych	111
Kryptografia	114
Podpis elektroniczny	119
Elementy netykiety	120
Elementy ergonomii stanowiska pracy komputerowej	122
VIII. BAZY DANYCH	125
Wprowadzenie	125
Systemy baz danych	128
Przykładowa baza danych	132
Język manipulacji i zapytań	136
Hurtownie danych	139
IX. ZASTOSOWANIA INFORMATYKI W GOSPODARCE	143
Wstęp	143
Informatyka w przedsiębiorstwach	145
Informatyka w bankach	146
Usługi i handel elektroniczny	147
Społeczeństwo informacyjne	148
X. PRZEDSIĘWZIĘCIA INFORMATYZACYJNE	151
Informatyzacja	151
Cykl życia oprogramowania	155
LITERATURA	161

PRZEDMOWA

Pełny tytuł książki powinien brzmieć *Elementy informatyki dla nieinformatyków*. Książka jest bowiem przeznaczona dla studentów pierwszych lat kierunków nieinformatycznych, którzy mają wprawdzie pracować w zawodach wykorzystujących informatykę, ale nie zajmujących się jej rozwijaniem.

Przy pisaniu tego rodzaju książki zasadniczy jest problem zakresu i szczegółowości prezentacji wybranego materiału. Wynika to z dwóch powodów: po pierwsze, wiedza i umiejętności absolwentów szkół średnich są bardzo zróżnicowane; po drugie, informatyka, rozwijając się szybko i intensywnie, powoduje, że wyłaniają się nowe zagadnienia, a dotychczasowe się dezaktualizują.

Dobór tematów omawianych w książce wynika z pragmatycznego widzenia informatyki jako dyscypliny naukowo-technicznej, która dostarcza specyficznych metod i narzędzi rozwiązywania problemów. Można wyróżnić trzy zasadniczo różne sytuacje, z którymi może spotkać się nieinformatyk wykorzystujący informatykę.

Najbardziej powszechna sytuacja wiąże się z codziennym, standardowym wykorzystaniem komputera osobistego, zwykle podłączonego do Internetu, do szeroko rozumianych prac biurowych, obejmujących m.in. tworzenie i przechowywanie dokumentów, prowadzenie korespondencji i wyszukiwanie informacji w Internecie.

Druga sytuacja wiąże się z zastosowaniem informatyki w pracy zawodowej: w przedsiębiorstwie, banku, jednostce administracji państwowej lub samorządowej, gdzie używa się specjalistycznego oprogramowania.

Są także sytuacje, gdy użytkownicy informatyki podczas pracy zawodowej stwierdzają, że stosowane środki informatyczne są niewystarczające lub dostrzegają obszary działania, które można usprawnić, wykorzystując informatykę. W takich przypadkach — jako niespecjaliści — powinni zwrócić się po pomoc do informatyków (mając elementarną wiedzę na

temat przedsięwzięć informatycznych, mogą nawiązać z nimi efektywną współpracę).

Podstawowa wiedza w każdym z wymienionych obszarów jest, zdaniem autora, przydatna absolwentowi kierunku nieinformatycznego.

Książka w zasadzie koncentruje się na przekazaniu podstawowych informacji z pierwszego zakresu. Zagadnienia te omawia się w zwarty sposób, kładąc nacisk na wyjaśnienie znaczenia podstawowych pojęć.

Drugi i trzeci zakres są tylko zarysowane. Wynika to ze złożoności współczesnych aplikacji. Ich zrozumienie i nabycie umiejętności posługiwania się nimi wymaga, nawet od informatyka, oddzielnego szkolenia.

Książka nie jest instrukcją posługiwania się komputerem ani jego podstawowym oprogramowaniem. Opanowanie tej umiejętności wymaga przede wszystkim ćwiczeń praktycznych. Książka nie omawia nawet podstawowych narzędzi programowania; zagadnieniom tym, np. edytorom tekstowym, arkuszom kalkulacyjnym, są poświęcone liczne podręczniki. Ich objętość wyjaśnia, dlaczego zagadnienia te nie zostały włączone do książki.

Motywacje dotyczące zakresu i stopnia szczegółowości prezentacji były inspirowane zakresem podstawowych pojęć technologii informatycznej, wymaganych na egzaminie teoretycznym na Europejskie Komputerowe Prawo Jazdy (ECDL — *European Computer Driving Licence*), rekomendowane przez Polskie Towarzystwo Informatyczne. Zostało ono opracowane przez Unię Europejską — z myślą o tych, którzy muszą lub chcą wiedzieć, jak korzystać z komputera — i jest uznawane w całej Europie. Certyfikat ECDL otrzymuje się na podstawie pomyślnego zdania egzaminu teoretycznego i sześciu egzaminów praktycznych sprawdzających umiejętność obsługi komputera.

W książce znajdują odzwierciedlenie wymagania na egzamin teoretyczny, które obejmują:

- podstawowe pojęcia technik komputerowych,
- programy użytkowe i zastosowania komputerów,
- techniki informatyczne i społeczeństwo,
- bezpieczeństwo, prawa autorskie, regulacje prawne,
- sprzęt i oprogramowanie systemowe, ergonomia,
- sieci informatyczne.

Poza zakresem książki pozostają zagadnienia wymagane na egzaminach praktycznych, które obejmują:

- użytkowanie komputerów — znajomość podstawowych funkcji porządkujących środowisko pracy każdego użytkownika komputera w celu zwiększenia efektywności jego wykorzystania;
- przetwarzanie tekstów — używanie komputera do tworzenia, edycji, formatowania, przechowywania i drukowania dokumentów. Większość obecnie używanych dokumentów jest tworzona i opracowywana przy użyciu programów przeznaczonych do tego celu;
- arkusze kalkulacyjne — podobne do zwykłych formularzy rachunkowych, umożliwiają szybkie wykonywanie obliczeń; używane są do przygotowywania budżetu, opracowywania prognoz, sporządzania wykresów i raportów finansowych;
- bazy danych — pomagają w organizowaniu dużych zasobów danych, umożliwiając szybki i łatwy dostęp do nich;
- grafika inżynierska i prezentacyjna — ważne narzędzie architektów, inżynierów, projektantów i menedżerów. W ostatnich latach wzrosło znaczenie stosowania wykresów i prezentacji w wielu dziedzinach jako bardzo efektywnego środka komunikacji, szeroko wykorzystywanego w biznesie i nauczaniu;
- usługi w sieciach informatycznych — zastosowanie sieci rozwinęło się z potrzeby wspólnego korzystania z zasobów i szybkiego komunikowania się z innymi użytkownikami komputerów; dzisiaj, gdy miliony komputerów na całym świecie są ze sobą połączone, ważne jest, by posiadacze ECDL umieli wydajnie korzystać z Internetu.

Dwa pierwsze rozdziały książki mają charakter ogólnych podstaw, omawia się tu podstawowe pojęcia, takie jak *informacja* i *algorytm*. Trzy kolejne dotyczą komputera osobistego. Wyjaśnia się w nich elementy składowe konfiguracji sprzętowej i podstawowego oprogramowania komputera oraz omawia komunikację użytkownika z komputerem. Rozdział VI dotyczy sieci komputerowych, objaśniając ich elementy składowe i podstawowe usługi. Natomiast rozdział VII omawia główne zasady bezpiecznej pracy w sieci. Rozdział VIII jest poświęcony bazom danych, które są nieodłącznym elementem wszelkich aplikacji. Wreszcie ostatnie dwa rozdziały omawiają wybrane zagadnienia związane z zastosowaniami informatyki i jej wpływem na społeczeństwo.

I. CZYM JEST INFORMATYKA?

Informacja

W ogólnym, encyklopedycznym określeniu *informatyka* jest rozumiana jako dziedzina naukowo-techniczna zajmująca się całokształtem zagadnień związanych z:

- pobieraniem,
- przechowywaniem,
- przesyłaniem,
- przetwarzaniem,
- interpretowaniem informacji.

Wymienione czynności skojarzone z informacją, jej obiegiem i przetwarzaniem nie są wyłączną domeną informatyki. Z informacją mamy do czynienia w wielu sytuacjach życiowych: w działalności instytucji i organizacji, w systemach technicznych czy organizmach żywych. Specyfika informatyki polega na tym, że do wykonywania wymienionych czynności wykorzystuje komputery jako pewną klasę urządzeń technicznych.

Przedmiotem informatyki jest informacja. W ujęciu encyklopedycznym *informacja* to każdy czynnik zmniejszający stopień niewiedzy (nieokreśloności) o badanym zjawisku, umożliwiający człowiekowi, organizmowi żywemu lub urządzeniu automatycznemu polepszenie znajomości otoczenia i umożliwiający sprawniejsze przeprowadzanie celowego działania.

Źródłem informacji są odbierane wiadomości. Wpływają one na wiedzę odbiorcy, i to w różnym stopniu — nawet ta sama wiadomość może, w różnych okolicznościach, mieć różny wpływ na wiedzę tego samego odbiorcy. Mówimy, że w odebranej wiadomości jest zawarta informacja, a miarą ilości informacji jest wpływ na wiedzę odbiorcy.

Wiedzę odbiorcy w danej dziedzinie można określić jako znajomość odpowiedzi na pewien zbiór pytań. Stąd określenie jednostki ilości informacji wiąże się z oceną wartości odpowiedzi na pewien typ pytań — ta-

kich, na które można odpowiedzieć tylko *tak* albo *nie*, przy czym każda z tych odpowiedzi jest jednakowo prawdopodobna. Jednostką ilości informacji jest *bit*. Zdobywamy 1 bit informacji, np. gdy usłyszymy odpowiedź na pytanie:

Czy moneta upadła na stronę awersu?

Zakładając, że szanse upadku monety na którąkolwiek stronę są jednakowe, uzyskana ilość informacji wynosi 1 bit i jest niezależna od tego, czy usłyszymy odpowiedź *tak* czy *nie*.

Natomiast w przypadku pytania:

Czy kulka w ruletce zatrzymała się na polu 23 (jednym z 37 pól)?

odpowiedź *tak* daje nam więcej informacji niż odpowiedź *nie*. Odpowiedź *tak* określa bowiem dokładne położenie kulki, natomiast na podstawie odpowiedzi *nie* wiemy tylko, że pozostaje na jednym z 36 pozostałych pól.

W obu przypadkach zakładaliśmy, że pytający, zadając pytanie, nie wiedział, na którą stronę upadła moneta ani na którym polu zatrzymała się kulka w ruletce. Gdyby bowiem to wiedział, odpowiedzi nie stanowiłyby dla niego żadnej nowej informacji.

Podobny przykład stanowi sytuacja, gdy otrzymujemy od przyjaciela z innego miasta SMS o treści: *Przyjeżdżam jutro*, a po chwili kopię tę samej wiadomości. Pierwsza wiadomość zmienia naszą wiedzę o tym, co ma zdarzyć się następnego dnia, natomiast jej powtórzenie naszej wiedzy już nie zmienia.

W pytaniu o monetę każda z odpowiedzi dostarczała nam informacji w ilości 1 bitu, a ile informacji otrzymamy w przypadku pytania z ruletką?

Zakładając, że ruletka jest uczciwa, to znaczy, że wyrzucona kulka z jednakowym prawdopodobieństwem zatrzyma się na jednym z 37 pól, prawdopodobieństwo otrzymania odpowiedzi *tak* wynosi $1/37 \approx 0,027$, a odpowiedzi *nie* — $36/37 \approx 0,73$. Zauważmy, że otrzymanie odpowiedzi mniej prawdopodobnej daje nam więcej informacji niż otrzymanie odpowiedzi bardziej prawdopodobnej.

Przyjętą miarę liczbową ilości informacji uzyskanej przez otrzymanie odpowiedzi o prawdopodobieństwie p wyznacza wzór:

$$-\lg_2 p$$

gdzie $\lg_2$ jest logarytmem o podstawie 2. Ponieważ prawdopodobieństwo p jest liczbą z zakresu $(0,1)$, a więc logarytm o podstawie 2 z p jest liczbą ujemną, znak minus przed całym wyrażeniem powoduje, że otrzymana w ten sposób miara jest liczbą dodatnią. Zatem w przypadku odpowiedzi

tak ilość informacji, jaką otrzymujemy, wynosi $-\lg_2(1/37) \approx 5,21$ bitów, natomiast w przypadku odpowiedzi *nie* wynosi $-\lg_2(36/37) \approx 0,039$ bita.

Sprawdźmy, jakie wyliczenia daje podany wzór dla szczególnych przypadków.

Jeżeli otrzymujemy odpowiedź *tak* na pytanie dotyczące monety, to ilość otrzymanej informacji wynosi:

$$-\lg_2(1/2) = \lg_2 2 = 1 \text{ bit}$$

gdyż prawdopodobieństwo odpowiedzi *tak* wynosi $1/2$.

Jeżeli otrzymujemy odpowiedź *tak* na pytanie:

Czy zachodzi wskazana sytuacja spośród n jednakowo prawdopodobnych czterech sytuacji?

to odpowiedź *tak* daje nam informację w ilości:

$$-\lg_2(1/n) = \lg_2 n \text{ bitów.}$$

Dla $n = 4$ oznacza to 2 bity, dla $n = 8$ oznacza to 3 bity, dla $n = 16$ oznacza to 4 bity itd.

Dla zaspokojenia ciekawości zapytajmy jeszcze: ile informacji zdobywamy, wysłuchując komunikatu o losowaniu totolotka? Gra polega na wyborze 6 z 49 liczb. Prawdopodobieństwo wylosowania pierwszej z 6 liczb wynosi $6/49$, drugiej — $5/48$, trzeciej — $4/47$, ... i ostatniej, szóstej — $1/44$. Łączne prawdopodobieństwo jest iloczynem tych prawdopodobieństw i wynosi około:

$$7,15 \cdot 10^{-8}$$

Z tego wynika liczba bitów informacji:

$$-\lg_2(7,15 \cdot 10^{-8}) \approx -23,73$$

Przykład ilustruje, jaki ogrom reprezentuje kilkadziesiąt bitów informacji.

Podsumujmy: o informacji mówimy wówczas, gdy jest odbiorca, do którego docierają pewne wiadomości (komunikaty, odpowiedzi na zadane pytania), a ilość informacji, jaka jest zawarta w tych wiadomościach, nie jest określona bezwzględnie, ale zależy od wiedzy odbiorcy.

W komunikacji między ludźmi wiadomości, a więc pośrednio i informacja, są przekazywane w różnej postaci, np. mowy, pisma, gestów czy mimiki. W przypadku komunikacji ludzi z urządzeniami technicznymi, w tym z komputerami, przekazywane wiadomości są reprezentowane również w innych postaciach: tekstowej (znakowej), graficznej, akustycznej (audio), obrazu ruchomego (video).

Komputery odbierają od swoich użytkowników wiadomości oraz gromadzą je, przechowują i przetwarzają. Przetwarzanie oznacza, że na podstawie zgromadzonych i otrzymywanych wiadomości są opracowywane i zapamiętywane bądź przekazywane do użytkowników komputera nowe wiadomości. Stwierdzenie to, dosyć oczywiste, wskazuje na to, że komputery bezpośrednio nie operują na informacji, gdyż informacja wiąże się z interpretacją wiadomości, które otrzymuje użytkownik komputera, ale na wiadomościach. Chociaż często używa się określenia, że komputery przetwarzają informację, to lepiej mówić, że przetwarzają one dane.

Dane

Pojęcie *danej* jest ogólniejsze niż pojęcie *wiadomości*, gdyż z pojęciem wiadomości kojarzy się zwykle nadawcę i odbiorcę, podczas gdy przechowywanym wiadomościom trudno bezpośrednio przypisać ich nadawców lub odbiorców. Stąd np. w przypadku opisu działania programu mówimy, że na podstawie danych wejściowych program prowadzi obliczenia, dostarczając danych wynikowych. Mówimy też o bazach danych i o wyszukiwaniu w bazach danych.

Ogólnie przez *daną* rozumie się pewien nazwany atrybut (cechę) oraz związaną z nią wartość. Wartość ta może być prosta (nierozkładalna na elementy składowe) lub złożona. Stąd wyróżnia się dane proste i dane złożone.

Przez *daną prostą* rozumie się parę elementów: nazwa danej i związaną z nią wartość prostą, np.:

Nazwisko: 'Kowalski'

gdzie *Nazwisko* jest nazwą danej, a 'Kowalski' — ciągiem znaków stanowiących wartość danej, symbol dwukropka został użyty na rozdzielenie obu elementów danej. Podobnie danymi prostymi są pary:

Nazwisko: 'Nowak'

Nazwisko: 'Smith'

czy też inne rodzaje par:

Wiek: 44

MiejsceUrodzenia: 'Wrocław'

RodzajPrzedsiębiorstwa: 'Spółka z o.o.'

Zbiór wartości, jaki może być związany z daną o pewnej nazwie, określa się jako *dziedzinę* danej. O danych prostych mówimy, że są tego samego typu, gdy mają taką samą nazwę i taką samą dziedzinę.

Dane złożone powstają przez łączenie (składanie) danych prostych. Podstawowymi sposobami składania danych (prostych i złożonych) są konstrukcje *ciągu* i *krotek* (rekordów). Przykładem danej złożonej, która powstaje przez utworzenie ciągu z danych prostych tego samego typu, jest ciąg o nazwie *ListaNazwisk*, składający się z trzech danych prostych:

ListaNazwisk: (*Nazwisko*: 'Kowalski'; *Nazwisko*: 'Nowak'; *Nazwisko*: 'Smith')

ListaNazwisk jest nazwą danej złożonej, a wartością danej jest ciąg: (*Nazwisko*: 'Kowalski'; *Nazwisko*: 'Nowak'; *Nazwisko*: 'Smith').

Symbole nawiasów okrągłych służą do oznaczenia początku i końca ciągu, a średniki — do oddzielenia kolejnych elementów ciągu.

Krotki są ciągami o ustalonej długości, których elementami mogą być dane dowolnych, niekoniecznie jednakowych typów. Przykładem danej złożonej powstałej przez tworzenie krotek jest trójka danych różnego typu:

DaneOsobowe: <*Nazwisko*: 'Kowalski'; *Wiek*: 44; *MiejsceUrodzenia*: 'Wrocław'>

DaneOsobowe są tu nazwą, a wartością danej jest 3-krotka (uporządkowana trójka):

<*Nazwisko*: 'Kowalski'; *Wiek*: 44; *MiejsceUrodzenia*: 'Wrocław'>

Symbole nawiasów trójkątnych służą do oznaczenia początku i końca ciągu, a średniki — do oddzielenia kolejnych elementów krotki.

Dane złożone mogą powstawać przez wielokrotne zastosowanie różnych sposobów składania danych. Wyjaśnia to przykład:

ListaDanychOsobowych:

(*Osoba_1*: <*Nazwisko*: 'Kowalski'; *Wiek*: 44; *MiejsceUrodzenia*: 'Wrocław'>;

Osoba_2: <*Nazwisko*: 'Nowak'; *Wiek*: 33; *MiejsceUrodzenia*: 'Zgierz'>;

Osoba_3: <*Nazwisko*: 'Smith'; *Wiek*: 22; *MiejsceUrodzenia*: 'Londyn'>)

Dana złożona o nazwie *ListaDanychOsobowych* ma wartość:

(*Osoba_1*: <*Nazwisko*: 'Kowalski'; *Wiek*: 44; *MiejsceUrodzenia*: 'Wrocław'>;

Osoba_2: <*Nazwisko*: 'Nowak'; *Wiek*: 33; *MiejsceUrodzenia*: 'Zgierz'>;

Osoba_3: <*Nazwisko*: 'Smith'; *Wiek*: 22; *MiejsceUrodzenia*: 'Londyn'>).

Użyte wyżej oznaczenia mają charakter lokalny, służą tylko wyjaśnieniu, czym są dane. Sposoby zapisu danych w językach programowania czy w bazach danych są inne.

Kodowanie danych

Specyficzną własnością komputerów jest uniwersalny sposób reprezentacji danych — każda dana jest reprezentowana w postaci ciągu zero-jedynkowego (binarnego). Przyjęty sposób wynika z możliwości technicznych. Fizyczną reprezentacją zer i jedynek w komputerze lub w jego pamięci są zwykle impulsy elektryczne lub obszary namagnesowania. Rozróżnienie dwóch poziomów napięcia czy stanów namagnesowania jest łatwiejsze do odróżnienia od trzech lub więcej poziomów, a ponadto biorąc pod uwagę to, że fizyczne wielkości mogą zmieniać się w czasie, odróżnianie dwóch różnych stanów jest bardziej niezawodne niż odróżnianie trzech lub więcej stanów. W rezultacie obecnie niemal powszechnie jest stosowane kodowanie binarne danych.

Zasady kodowania wyjaśnimy na przykładzie kodowania znaków. Załóżmy, że mamy do dyspozycji zbiór symboli — cyfr arabskich, służący do zapisu liczb dziesiętnych:

$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Zasada kodowania zakłada, że do reprezentacji każdej cyfry przyjmujemy ciąg binarny, a ciągi reprezentujące różne cyfry mają tę samą długość. Ponieważ mamy 10 różnych symboli, minimalną długością ciągu binarnego jest 4. Ciąg binarny o długości 4 daje 16 różnych kombinacji, ale ciąg o długości 3 daje tylko 8 kombinacji. Łatwo sprawdzić, że ciąg binarny o długości n wyznacza 2^n różnych kombinacji zero-jedynkowych.

Przykładowe przyporządkowanie, jakie moglibyśmy zdefiniować, wyraża 10 pierwszych wierszy tab. 1.1.

Tabela ta liczy 16 wierszy, gdyż tyle jest różnych ciągów binarnych o długości 4, pozostałe wiersze można więc wykorzystać do zakodowania innych symboli, np. symbolu spacji i przecinka, jak to uczyniono w wierszach 11 i 12.

Przy takich zasadach kodowania liczbę dziesiętną 23 będzie kodować ciąg binarny:

0010 0011

Odstęp pomiędzy kolejnymi czwórkami zachowano tylko w celu lepszego czytelności. Liczbę 23,34 zapiszemy jako ciąg binarny:

0010 0011 1011 0011 0100

natomiast ciąg dwóch liczb: 23,34 21,9 oddzielanych spacją można reprezentować ciągiem binarnym:

0010 0011 1011 0011 0100 1010 0010 0001 1011 1001

Pojedynczą pozycję ciągu binarnego nazywa się *bitem*. W przykładzie kodujemy symbole cyfr ciągami 4-bitowymi.

Tab. 1.1
Przykład kodowania symboli

Lp.	Symbol	Kod binarny
1	0	0000
2	1	0001
3	2	0010
4	3	0011
5	4	0100
6	5	0101
7	6	0110
8	7	0111
9	8	1000
10	9	1001
11	spacja	1010
12		1011
13		1100
14		1101
15		1110
16		1111

Należy zwrócić uwagę, że termin bit ma w tym przypadku zupełnie inne znaczenie niż ten sam termin wprowadzony wcześniej i rozumiany jako jednostka ilości informacji!

Pojęcia *bitu* używa się tutaj do określenia długości ciągów binarnych, mówimy o ciągach czy kodach n -bitowych.

Ale pojęcia bitu używa się także do innych celów, np. do określenia pojemności pamięci komputerów. Ponieważ bit jest małą jednostką, stosuje się, podobnie jak w przypadku wielkości fizycznych, wielokrotności tej jednostki. Wielokrotnościami są potęgi liczby 2, a nie potęgi liczby 10.

Podstawową, używaną wielokrotnością jest 1 Kbit = $2^{10} = 1024$. Ze względu na bliskość liczby 1024 do 1000, przyjęto pisać 1 Kb (duża litera K) i czytać jako jeden kilobit, tak samo jak np. 1 kg — jeden kilogram. Większymi jednostkami są 1 Mb = $(1 \text{ Kb})^{10}$, 1 Gb = $(1 \text{ Mb})^{10}$, 1 Tb = $(1 \text{ Gb})^{10}$ itd. Przedrostki K, M, G są zatem używane jako skróty na oznaczenie liczb 2^{10} , 2^{20} , 2^{30} .

Obok bitu bardzo często posługujemy się pojęciem *bajtu* lub *oktetu*. Jeden bajt lub oktet (1 B), oznacza ciąg binarny o długości 8 bitów. Dla

bajtu używamy także określonych wyżej wielokrotności, pisząc np. 1 KB, 1 MB, 1 TB. Przy czym 1 KB to 8×2^{10} b, 1 MB to 8×2^{20} bitów.

Przedstawiony przykład ilustruje zasadę kodowania, ale nie pokazuje rzeczywistej złożoności problemu kodowania używanych symboli.

Zbiory znaków stosowane w informatyce są znacznie większe od zbioru symboli cyfr arabskich. Zbiory takie powinny zawierać wszystkie znaki stosowane w piśmie danego kraju. W różnych krajach stosuje się różne alfabety. W krajach europejskich dominuje alfabet łaciński, przy czym w Polsce, Niemczech, Czechach, Szwecji, Danii, Francji czy Hiszpanii stosowane są pewne odmiany wyróżniające się stosowaniem znaków diakrytycznych (ozdobników) do liter alfabetu łacińskiego, jak: $\grave{a}$, $\grave{a}$, $\acute{a}$, $\hat{a}$, $\tilde{a}$, $\ddot{a}$, $\text{\AA}$, $\text{\textcircled{a}}$, $\text{\textcircled{e}}$, $\text{\textcircled{e}}$, $\text{\textcircled{e}}$, $\text{\textcircled{e}}$, $\text{\textcircled{e}}$, $\text{\textcircled{e}}$. Oprócz alfabetu łacińskiego na świecie w użyciu są inne alfabety: grecki, arabski, hebrajski, japoński czy chiński.

Poza literami w piśmie używa się jeszcze innych symboli: cyfr, znaków interpunkcyjnych, symboli matematycznych. W edycji tekstów wyróżnia się dodatkowo inne znaki, zwane znakami sterującymi, które nie mają bezpośredniej reprezentacji tekstowej, ale które służą np. do adiuśtacji tekstu. Znakami tymi są m.in.: koniec wiersza, znak tabulacji, koniec strony, znak przypisu dolnego lub górnego. Używa się także specyficznych znaków, które mają znaczenie przy edycji tekstów lub podczas transmisji danych, m.in. znaki sterowania kursorem, znaki określające początek i koniec transmitowanego ciągu znaków, znak potwierdzenia transmisji.

Bogactwo stosowanych symboli doprowadziło do międzynarodowych porozumień dotyczących ustalenia repertuarów używanych symboli i jednolitego sposobu ich kodowania. Zbiór znaków i przyporządkowanych im kodów binarnych przyjęto nazywać kodem znakowym. Jednym ze starszych, szeroko rozpowszechnionych kodów znakowych jest 8-bitowy (1-bajtowy) kod ASCII (*American Standard Code for Information Interchange*); jego europejskim odpowiednikiem jest kod Latin 1. Kody te pozwalają na bezpośrednią reprezentację tylko $2^8 = 256$ różnych znaków. Obecnie jednak wychodzą one z użycia, a ich miejsce zajmują inne, np. 16-bitowy (2-bajtowy) kod *Unicode* czy też rozwijany pod patronatem Międzynarodowej Organizacji Standaryzującej (ISO — *International Standard Organization*), również 16-bitowy, kod BMP (*Basic Multilingual Plane*). Kody 16-bitowe pozwalają na bezpośrednią reprezentację $2^{16} = 2^6 \times 2^{10} = 64 \text{ K}$ różnych znaków.

Przedstawione informacje dotyczą tylko kodowania symboli. Oddzielnym zagadnieniem jest kodowanie danych w innych postaciach: obrazów, dźwięków, filmów (zob. rozdział IV).

System informacyjny i system informatyczny

Z terminem *informacja* spotykamy się często w wielu życiowych sytuacjach. Pytamy np.:

Czy mógłbym zostać poinformowany w sprawie...?

Słyszymy w odpowiedzi: niestety, nie mam informacji w tej sprawie. Równie często spotykamy się także z oceną ilości informacji, np. stwierdzając: to jest cenna informacja, albo: to jest informacja bezwartościowa. Raczej rzadko natomiast oceniamy wartość informacji w bitach. Ten sposób oceny ilości informacji jest przydatny w pewnych szczegółowych rozważaniach, np. w telekomunikacji.

Informacja jest ściśle związana z funkcjonowaniem przedsiębiorstwa, instytucji czy organizacji gospodarczej, społecznej. Np. dla Urzędu Miejskiego — jednostki administracji samorządowej — źródłami informacji są mieszkańcy, jednostki gospodarcze i społeczne działające na danym terenie, a także inne jednostki administracji, z którymi współpracuje lub którym podlega. Te same jednostki i petenci są także odbiorcami informacji przygotowywanej przez dany urząd. W obrębie urzędu pewne informacje są gromadzone, a także przetwarzane. W omawianym przykładzie w zasadzie wszystkie informacje mają formę pisemnych dokumentów.

Urząd możemy traktować jako przykład systemu informacyjnego. Mówiąc o Urzędzie Miejskim jako systemie informacyjnym, pomijamy wiele istotnych aspektów; abstrahujemy od ludzi, którzy tam pracują, od budynków, w którym się mieszczą. System informacyjny jest tylko modelem danej organizacji, który skupia się na pewnych aspektach. Pełny model powinien oczywiście określać cel funkcjonowania urzędu, wskazywać jego jednostki organizacyjne (wydziały, komisje) i ich wzajemne powiązania, a także jawnie definiować otoczenie urzędu.

Pojęcia *modelu* i *systemu* są używane w wielu sytuacjach.

Modelem nazywamy opis danego, interesującego nas wycinka rzeczywistości. Tworząc opis, czynimy to w pewnym celu, zatem koncentrujemy się tylko na wybranych, interesujących nas aspektach i na ustalonym poziomie szczegółowości — mówimy, że budujemy model, patrząc z pewnej perspektywy.

System jest zbiorem wzajemnie powiązanych elementów, wyodrębnionych z otoczenia ze względu na te powiązania. Powiązania między elementami systemu tworzą jego strukturę. Systemowi przyświeca cel funkcjonowania i sprawia on zarazem, że system jest niepodzielną całością. Oznacza to, że każdy, dowolnie wyodrębniony podzbiór elementów ma wpływ na funkcjonowanie systemu, ale żaden z podzbiorów nie ma wyłącznego wpływu, czyli usunięcie dowolnego elementu nie pozwala pozostałym elementom na realizację celu systemu.

W omawianym przykładzie systemu informacyjnego jedynym interesującym nas aspektem była informacja, a dokładniej dokumenty, które wpływają i wypływają z urzędu, oraz te, które są w urzędzie gromadzone. Precyzując model, należałoby określić szczegółowość prezentacji dokumentów, np. czy chodzi tylko o ich nazwy i identyfikatory, czy o ich strukturę (spis treści), czy też o pełną ich treść.

Pojęcie systemu stało się przedmiotem teorii systemów — działu nauki, który wyłonił się z cybernetyki. Według teorii systemów w świecie otaczającym człowieka można rozróżnić:

- systemy rzeczywiste — techniczne, społeczne, biologiczne,
- pojęciowe — logika, matematyka,
- abstrakcyjne — język.

Systemem, w nieco innym znaczeniu, określa się również zespół sposobów (metod) działania, wykonywania złożonych czynności, a także całością zasad organizacji, ogół norm i reguł obowiązujących w danej dziedzinie (np. system finansowy, system moralny); także całościowy i uporządkowany zespół zdań powiązanych ze sobą określonymi stosunkami logicznymi (np. każda teoria metodologicznie poprawna i dotycząca dostatecznie obszernego fragmentu rzeczywistości).

Pojęcie *systemu informacyjnego* może stanowić zatem pewien model w zasadzie każdego systemu technicznego, społecznego czy gospodarczego. Tworzenie takiego modelu jest zwykle nieodłącznym etapem budowania systemu informatycznego.

System informatyczny jest zespołem środków technicznych i programowych, służącym do gromadzenia, przechowywania i przetwarzania informacji. W jego skład wchodzi komputery, często połączone w sieć, wraz z odpowiednim oprogramowaniem, a także różne urządzenia pomocnicze (np. drukarki, skanery, pamięci).

System informatyczny możemy określić jako specyficzny rodzaj systemu informacyjnego. Jego specyfika polega na tym, że obieguje i przetwarza-

nie informacji są realizowane, przynajmniej częściowo, przy użyciu środków technicznych.

Systemy informatyczne stały się nieodłącznym elementem życia współczesnych społeczeństw. Przykładem są liczne systemy stosowane w przedsiębiorstwach — od systemów przeznaczonych do wąskiej klasy zadań, takich jak systemy finansowo-księgowe, po systemy zintegrowanego zarządzania; od dziesiątków lat funkcjonują także systemy bankowe i systemy rezerwacji biletów lotniczych, a specjalną klasę stanowią informatyczne systemy militarne.

Nie zawsze w przypadku zastosowań informatyki mówimy o systemach informatycznych. Wykorzystanie komputera w pracy domowej, w pracy biurowej, obliczeniach inżynierskich nie wymaga posługiwania się tym terminem.

II. ALGORYTM, KOMPUTER, JĘZYK PROGRAMOWANIA

Algorytm

Przedmiotem informatyki jest informacja, a właściwie dane, które są nośnikiem informacji i które można przetwarzać za pomocą komputera. Komputer jest urządzeniem technicznym, które może samodzielnie prowadzić obliczenia, czyli wykonywać algorytmy zdefiniowane przez użytkownika.

Co to jest komputer? Co to jest algorytm? — oto pytania, które wymagają wyjaśnienia.

Pojęcia *algorytmu* nie daje się wprowadzić bez pewnej dyskusji. Nieformalnie algorytm to metoda udzielania odpowiedzi na pytanie z pewnej klasy pytań; klasę tych pytań określa się jako *problem*.

Rozumienie pytania i odpowiedzi zależy od tego, kto tych pojęć używa, ale zawsze będziemy zakładać, że zarówno pytanie, jak i odpowiedź muszą być wyrażone w tym samym, najlepiej pisanym języku. Dodatkowo zakładamy, że wiemy, czy coś jest, czy nie jest *poprawną* odpowiedzią na zadane pytanie.

Na dane pytanie może nie być odpowiedzi, może być tylko jedna, wiele, a nawet nieskończenie wiele odpowiedzi, np.:

Dla jakiej liczby rzeczywistej x zachodzi $x^2 + 25 = 36$?

Dla jakiej liczby rzeczywistej x zachodzi $x + 25 = 36$?

Dla jakiej liczby rzeczywistej x zachodzi $x^2 + 36 < 25$?

Odpowiedzią na pytanie pierwsze są dwie liczby, na drugie — jedna liczba, a na trzecie nie ma liczby stanowiącej odpowiedź.

Pytanie powinno być sformułowane w taki sposób, aby wskazywało jednoznacznie na to, co będzie dopuszczalną odpowiedzią. Np. pytanie:

Dla jakiej dodatniej liczby rzeczywistej x zachodzi $x^2 = 2x + 1$?

może budzić wątpliwości, czy dopuszczalną będzie odpowiedź $1 + \sqrt{2}$, czy też 2,414, gdyż liczba ta nie jest dokładnym pierwiastkiem równania.

W pytaniu należy dodatkowo określić, w jakiej postaci ma być wartość stanowiąca odpowiedź.

Na pytanie o liczbę określoną następująco:

Najmniejsza liczba, której najkrótszy opis wymaga użycia więcej niż dziesięciu słów.

nie ma odpowiedzi, gdyż liczba taka nie istnieje. Gdyby bowiem istniała, to powyższy opis, który ma jedenaście słów, byłby jej opisem.

Problem jest klasą pytań. Konkretnie pytanie z tej klasy jest *konkretyzacją* albo wystąpieniem problemu.

Np. dla pytania:

Dla jakiej liczby rzeczywistej x dla danego parametru a zachodzi $x^2 + a = 25$?

dopiero po ukonkretnieniu wartości a mamy do czynienia z pytaniem, na które można udzielić odpowiedzi.

W takim zakresie, w jakim problem jest interesujący, interesujące jest znalezienie odpowiedniego algorytmu. Wiąże się z tym podział na typy problemów. Większość problemów spotykanych w nauce i technice należy do jednego z dwóch typów:

- problemy funkcyjne (*zadania* obliczeniowe) — rozwiązanie takiego problemu polega na wyliczeniu wartości pewnej funkcji dla danego zestawu argumentów;
- problemy decyzyjne — rozwiązanie takiego problemu polega na udzieleniu odpowiedzi *tak* albo *nie* na zadane pytanie; typ ten może być wprowadzie uważany za szczególny przypadek pierwszego typu, ale często wygodnie typy te odróżniać.

Za najbardziej ogólny można uważać typ relacyjny — rozwiązaniem problemu, czyli odpowiedzią na zadane pytanie, może być zbiór odpowiedzi.

Algorytm prowadzi do rozwiązania danego problemu — jest zestawem czynności do udzielenia odpowiedzi na pytanie stanowiące ukonkretnienie danego problemu. Spełniają go następujące postulaty:

1. Algorytm jest zapisany w postaci skończonego napisu w pewnym języku naturalnym lub formalnym.

2. Konkretnie pytanie, na które należy udzielić odpowiedzi, jest jednoznacznie reprezentowane przez dane wejściowe algorytmu, również zapisane w pewnym języku.

3. Obliczenie (wykonanie) algorytmu jest sekwencją (ciągiem) kroków; każdy kolejny krok realizuje akcję obliczeniową, która albo produ-

kuje pewien wynik cząstkowy, albo stanowi decyzję o zakończeniu wykonania algorytmu. Krok jest realizowany w skończonym czasie i produkuje skończoną ilość danych.

4. Wynik cząstkowy wytworzony w danym kroku zależy tylko od:

- zapisu algorytmu,
- danych wejściowych,
- wyniku cząstkowego wytworzonego w poprzednim kroku.

5. Po zakończeniu wszystkich kroków odpowiedź na zadane pytanie stanowi jednoznacznie określoną część wyniku cząstkowego wytworzonego w ostatnim kroku.

6. Dla dowolnych danych wejściowych zakończenie wykonania następuje po skończonej liczbie kroków.

W podanym określeniu algorytmu zakłada się istnienie pewnego wykonawcy (mechanizmu), pozostawiając mu różne możliwości wykonania algorytmu. Komputer jest właśnie tym mechanizmem, który jest zdolny do wykonywania algorytmów, więcej: nie potrafi obliczyć nic, co nie byłoby określone przez pewien algorytm.

Sposób obliczenia algorytmu, czyli określenie kroków oraz wykonywanych akcji obliczeniowych, jest pozostawiony poza definicją; zależy to od wielu czynników, m.in. od zastosowanego formalizmu opisu algorytmów. Z akcją w obrębie kroku wiąże się tylko wymóg jednoznaczności i zrozumiałości dla wykonawcy algorytmu. W zależności od tego, czy wykonawcą akcji będzie człowiek, czy maszyna, zakres możliwych akcji obliczeniowych może być różny. Istotne jest natomiast to, że wykonanie akcji musi odbyć się w skończonym czasie, a jej wynikiem może być skończona liczba danych.

Zatem określenie algorytmu zależy od przyjętego poziomu abstrakcji w odniesieniu do tego, czym mogą być akcje obliczeniowe realizowane w oddzielnych krokach.

Postulat determinizmu (warunek 4) oznacza, że algorytm wykonywany przez dwóch różnych wykonawców da dokładnie ten sam ciąg kroków, z takimi samymi wynikami częściowymi, i taki sam wynik końcowy.

Warunek 6 wymaga zakończenia obliczeń po skończonej liczbie kroków.

Przykładem algorytmu, znanego już w starożytności, jest *algorytm Euklidesa* obliczania największego wspólnego dzielnika liczb całkowitych.

tych. Opis algorytmu, wyrażony w języku naturalnym, przedstawia się następująco:

Niech x_1, x_2 będą dwiema liczbami całkowitymi, dla których należy obliczyć największy wspólny podzielnik.

Obliczenie polega na konstrukcji ciągu liczb $x_1, x_2, \dots$. Załóżmy, że obliczyliśmy ciąg $x_1, x_2, \dots, x_i$, dla $i \geq 2$. Obliczenie kolejnego wyrazu ciągu x_{i+1} przebiega według kolejności:

1. jeżeli $x_i = 0$, to nie obliczaj dalszych liczb, zatrzymaj obliczenia i przyjmij wartość x_{i-1} jako końcowy wynik algorytmu,

2. jeżeli $x_{i-1} < x_i$, to przyjmij, że $x_{i+1} = x_{i-1}$,

3. jeżeli $x_i \leq x_{i-1}$, to przyjmij, że $x_{i+1} = x_{i-1} - x_i$.

Np. obliczenie największego wspólnego podzielnika liczb 352 i 154 prowadzi do konstrukcji następującego ciągu:

352, 154, 198, 154, 44, 110, 44, 66, 44, 22, 22, 0

z czego wynika, że poszukiwaną liczbą jest 22.

Mając dany problem, można zastanawiać się, czy istnieje dla niego algorytm, lub inaczej: czy dany problem daje się rozwiązać za pomocą komputera?

Problem, dla którego nie istnieje rozwiązanie w postaci algorytmu, nazywa się problemem algorytmicznie *nierozwiązywalnym*. Stwierdzenie, że dla danego problemu nie istnieje algorytm, jest czymś więcej niż stwierdzeniem, że nie znamy algorytmu dla tego problemu. W wielu bowiem przypadkach wiemy, że algorytm istnieje, chociaż możemy go nie znać.

Wiemy, że np. istnieje algorytm dla każdej skończonej klasy pytań, z których każde ma odpowiedź. Dla tego przypadku wiadomo też ogólnie, jak ma wyglądać algorytm — jest to wyczerpujący przegląd tabeli, której wierszami są pary: pytanie – odpowiedź. Nie oznacza to oczywiście, że dla dowolnej klasy pytań znamy taki algorytm.

Algorytm będzie mógł być skonstruowany tylko wówczas, gdy będą znane wszystkie pytania i adekwatne odpowiedzi w postaci pewnych napisów. Chociaż algorytm kończy się po skończonej liczbie kroków, to liczba ta może być tak duża, że praktycznie można uważać ją za nieskończoną. Dlatego praktycznie ważna jest znajomość *złożoności obliczeniowej* algorytmu. Złożoność obliczeniową określa się jako funkcję liczby operacji, które należy wykonać, aby uzyskać końcowy wynik, od rozmiaru problemu. Np. rozmiarem problemu znalezienia największej liczby w zbiorze n liczb jest liczebność tego zbioru, a liczba operacji porównania

dwóch liczb, które należy powtarzać w trakcie wyszukiwania jest proporcjonalna do n .

Obok pojęcia *algorytmu* często spotykamy się z podobnym pojęciem *procedury*. Procedurą (deterministyczną) dla danego problemu jest zestaw czynności, które spełniają warunki 1-5 definicji algorytmu.

Procedura udziela odpowiedzi na zadane pytanie, ale nie gwarantuje, że będzie zawsze się kończyć. Pojęcie procedury jest szersze niż pojęcie algorytmu, ale wiąże się z nim pewien praktyczny kłopot. Okazuje się, że próbując rozwiązać problemy, łatwo jest nam opracować pewną procedurę, ale czasami trudno stwierdzić, że obliczenia prowadzone zgodnie z opracowaną procedurą będą się kończyć po skończonej liczbie kroków dla dowolnych danych wejściowych. Prowadząc jakiegokolwiek obliczenia, chcemy otrzymać ich wynik w rozsądnym czasie. Jeżeli natomiast nie mamy gwarancji, że obliczenia kończą się w skończonym czasie, to tym bardziej nie mamy szansy na uzyskanie wyników w rozsądnym czasie. Okazuje się, że nie istnieje żadna metoda, która pozwalałaby na stwierdzenie, że dana procedura jest algorytmem, czyli że dla dowolnych danych wejściowych kończy się po skończonej liczbie kroków.

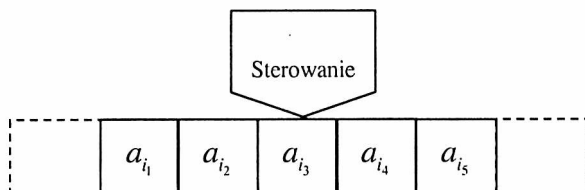
Maszyna Turinga¹

Informatyka dostarcza narzędzi i metod rozwiązywania wielu problemów, ale tylko takich, które są rozwiązywalne algorytmicznie. Stwierdzenie to wynika z możliwości komputera — podstawowego narzędzia informatyki. Jest pewnym paradoksem, że ustalenia tego, co komputer może obliczyć, dokonano, zanim powstał pierwszy komputer. Powszechnie za czas narodzin pierwszego komputera uznaje się drugą połowę lat czterdziestych i wiąże z nazwiskiem wybitnego amerykańskiego matematyka pochodzenia węgierskiego, Johna von Neumanna. Natomiast za pierwszy teoretyczny model komputera — a dokładniej algorytmu — uważa się tzw. maszynę Turinga, którą zaproponował brytyjski matematyk Allan Turing, w 1936 r. Pojęcia algorytmu i komputera można w tym momencie uważać za tożsame. Warto przypomnieć, że sam termin algorytmu wywodzi się od zlatynizowanej formy (*Algorismus*, *Algorithmus*) przydomka persko-arabskiego matematyka z IX w. Muhammeda ibn Musy.

¹ Przy pierwszym czytaniu podrozdział ten można pominąć; warto do niego powrócić po przeczytaniu całego rozdziału.

Turing przedstawiał swą maszynę następująco: składa się ona z dwóch części — pierwsza część modeluje pamięć, druga — jednostkę sterującą. Pamięć jest wyobrażona w postaci taśmy nieograniczonej długości, podzielonej na tzw. pola. W pojedynczym polu można zapisywać pojedyncze symbole, np. litery alfabetu. Liczba różnych symboli jest skończona. Taśma służy do zapisywania danych początkowych do obliczeń oraz do zapisywania wyników obliczeń.

Rys. 2.1
Maszyna Turinga



Jednostka sterująca jest połączona z taśmą za pomocą głowicy czytająco-piszącej. W danym momencie głowica znajduje się przy jednym polu taśmy, a jednostka sterująca jest w pewnym określonym stanie. Liczba takich stanów jest skończona. Jednostka odczytuje symbol, który znajduje się pod jej głowicą czytająco-piszącą. Po przeczytaniu symbolu wykonuje trzy następujące czynności, które zależą od aktualnego stanu jednostki i od przeczytanego symbolu:

- zmienia swój stan bieżący na nowy stan,
- w odczytane pole wpisuje nowy symbol,
- przesuwa głowicę czytająco-piszącą w lewo lub w prawo na sąsiednie pole albo przestaje się przesuwać i zatrzymuje dalsze działania.

Po wykonaniu tego zestawu czynności, czyli pojedynczego kroku obliczeniowego, maszyna wykonuje według tego samego schematu następne kroki obliczeniowe.

Obliczenie maszyny rozpoczyna się w jakiejś sytuacji początkowej, gdy na taśmie jest zapisany skończony ciąg symboli, głowica czytająco-pisząca jest ustawiona na wybranym polu, a jednostka sterująca znajduje się w ustalonym stanie początkowym. Obliczenie kończy się po wykonaniu pewnej liczby kroków obliczeniowych, po których nie następuje już dalszy przesuw głowicy. Ciąg symboli zapisany na taśmie reprezentuje wyniki końcowe obliczenia maszyny.

Mimo skrótości opisu przedstawia on schemat działania tak złożonego urządzenia jak komputer.

Turing postawił *hipotezę*, że każdy problem, o którym sądzimy, że da się rozwiązać *algorytmicznie*, to znaczy na drodze mechanicznego manipulowania na symbolach, można rozwiązać za pomocą odpowiedniej maszyny Turinga. Inaczej: dla każdego problemu, o którym sądzimy, że jest algorytmicznie rozwiązywalny, daje się zbudować odpowiednią maszynę Turinga.

Hipotezy Turinga nie można oczywiście udowodnić, gdyż pojęcie problemu algorytmicznie rozwiązywalnego jest tu rozumiane intuicyjnie. Hipotezę tę można tylko obalić, gdyby przedstawić przykład takiego problemu, który da się rozwiązać, ale nie za pomocą maszyny Turinga. Dotychczasowe doświadczenie potwierdza hipotezę Turinga i nie ma żadnych przesłanek, które pozwalałyby na przypuszczenie, że może być inaczej.

Poparcie dla hipotezy Turinga wynika także z przesłanek teoretycznych: niezależnie od maszyny Turinga powstały także inne modele obliczeniowe i okazało się, że wszystkie są równoważne maszynie Turinga.

Czy istnieją problemy, dla rozwiązania których nie można skonstruować maszyny Turinga? Tak, istnieją takie problemy, nazywamy je *problemami nierozstrzygalnymi*. Okazuje się, że problemów nierozstrzygalnych nie daje się rozwiązać nie tylko przy pomocy maszyny Turinga, ale także w jakikolwiek inny sposób.

Maszyna Turinga jest tylko modelem teoretycznym. Jej użycie do rozwiązywania praktycznych zadań byłoby niezmiernie uciążliwe. Dziś powiedzielibyśmy, że dana maszyna Turinga jest komputerem stałoprogramowym, to znaczy służy do rozwiązywania tylko jednego konkretnego problemu. Pierwszą uniwersalną maszyną cyfrową jest maszyna von Neumanna. Istota pomysłu von Neumanna — pomysłu, który dzisiaj uważamy za oczywisty — było umieszczenie scenariusza pracy maszyny (programu) w jej pamięci. Praca maszyny von Neumanna polega na odczytywaniu i wykonywaniu kolejnych poleceń tego scenariusza.

Człowiek czy komputer?

Warto postawić pytanie: jak możliwości obliczeniowe komputera przystają do możliwości intelektualnych człowieka? Zwrócimy przy tym uwagę na pewne rzeczy, których współczesne komputery nie są w stanie wykonać. Rozpatrzmy dwa przykłady.

Współczesne komputery są zdolne do natychmiastowej analizy niezmiernej ilości danych pochodzących z tomografii, czyli wielu zdjęć rentgenowskich np. ludzkiej głowy, wykonywanych pod stopniowo zwiększającym się kątem. Przeanalizowanych danych używa się do utworzenia przestrzennego obrazu mózgu pacjenta. Ten obraz dostarcza informacji o strukturze tkanek, a zatem umożliwia dokładne umiejscowienie wszelkich nieregularności, jak guzy czy nadmiar płynów. Uderzające jest to, że żaden z obecnych komputerów nie jest w stanie przeanalizować pojedynczego zdjęcia twarzy człowieka i określić jego wieku z dokładnością, powiedzmy, do 5 lat. A jednak większość kilkunastoletnich dzieci potrafi to zrobić.

Dzisiejsze komputery potrafią wspaniale rozgrywać partie szachów, pokonując nawet arcymistrzów. A przecież wystarczy zmienić nieznacznie reguły gry, a komputer stanie się bezradny, podczas gdy szachista-amator będzie w stanie, w miarę szybko, opanować nowe reguły i zagrać całkiem dobrze.

Może powyższe przykłady są niewłaściwe tylko dlatego, że nie potrafimy zbudować odpowiednio dobrego oprogramowania? Zapytajmy więc inaczej: czy komputer może myśleć?

Ktoś raz powiedział, że pytanie, czy komputery potrafią myśleć, można porównać z pytaniem, czy łodzie podwodne potrafią nurkować. Analogia jest całkiem na miejscu. Chociaż zdajemy sobie sprawę z możliwości łodzi podwodnych — i wiemy, że mogą one robić takie rzeczy, które przypominają nurkowanie — to jednak rzeczywiste nurkowanie jest czymś, co wiążemy z obiektami natury organicznej, takimi jak ptaki czy ludzie, a nie z łodziami podwodnymi. Podobnie rzeczywiste myślenie wiąże się w naszym rozumieniu z istotami ludzkimi, a być może z zachowaniami małąp i delfinów, ale nie ze zbiorem bitów i bajtów.

Dziedziną badań najbardziej związaną z tą problematyką jest sztuczna inteligencja. Z punktu widzenia laika wydawałoby się, że samo pojęcie jest sprzecznością terminologiczną. Jesteśmy skłonni uważać inteligencję za naszą cechę w istocie swojej nieprogramowalną, a więc niealgorytmizowalną. Dla wielu ludzi już sama idea inteligentnej maszyny brzmi niewłaściwie.

Wydaje się, że maszyna, o której twierdzi się, że jest inteligentna, musi wykazać się zachowaniem inteligentnym podobnym do człowieka. Nie wymagamy przy tym, by chodziła, mówiła, lecz by wnioskujeła i odpowiadała jak człowiek. Ponadto cokolwiek się przyjmie za kryterium inteligencji, ktoś powinien sprawdzić, czy maszyna je spełnia. Dochodzimy

do wniosku, że maszynę będzie można uznać za inteligentną, jeżeli potrafi przekonać przeciętnego człowieka, iż w kategoriach jego intelektu nie różni się od innego człowieka.

Takie właśnie rozumowanie jest podstawą *testu Turinga*. Polega on na zadawaniu komputerowi pytań. Zakładamy, że komputer znajduje się w pomieszczeniu innym niż osoba zadająca pytania. Pytający porozumiewa się z komputerem za pośrednictwem klawiatury i ekranu. Pytania i odpowiedzi są zapisywane w języku naturalnym. Pomijamy tu, dla uproszczenia, samo rozpoznawanie zdań języka naturalnego i formułowanie odpowiedzi. Jeżeli na podstawie analizy udzielonych odpowiedzi człowiek nie jest w stanie odróżnić, czy odpowiedzi udzielił komputer czy inny człowiek, to zachowanie komputera może uznać za inteligentne. Rozpatrzmy kilka przykładów pytań i problemy, jakie wiążą się z zaprogramowaniem komputera przy konstrukcji odpowiedzi.

Czy jesteś komputerem?

Odpowiedź na to pytanie jest bardzo prosta — brzmi oczywiście *nie*, ale na takiej podstawie nikt nie da się przekonać, że nie ma do czynienia z komputerem.

Która godzina?

Komputer musi mieć wbudowany zegar, który i tak ma większość komputerów, i bez trudu udzieli właściwej odpowiedzi.

Kiedy zamordowano prezydenta Narutowicza?

Odpowiedź na to pytanie wymaga pewnego zasobu wiedzy historycznej, ale i tu odpowiedź jest stosunkowo prosta.

Ile jest 24143218 razy 4312489?

Pozornie pytanie dla komputera jest banalne, ale gdyby odpowiedział na nie natychmiast, bez wątpienia pozwoliłby się zdekonspirować.

Czy białe mogą dać mata w jednym posunięciu w następującej sytuacji:...?

Odpowiedź na pytanie wymaga, by komputer miał pewną wiedzę o zasadach gry w szachy. Wiemy, że programy takie istnieją i grają wspaniale.

Opisz swoich rodziców.

Tu odpowiedź jest trudniejsza. Można wprawdzie przygotować z góry opis rodziców, ale pytający może zadać dodatkowe pytanie o niespodziewany szczegół, które łatwo zdradzi komputer.

Jakie wrażenie zrobił na tobie wiersz Wisławy Szymborskiej...?

Co myślisz o prezydenturze Lecha Wałęsy?

Jaka jest twoja opinia o realizacji przedwyborczych obietnic rządzącej koalicji?

Ostatnie trzy pytania są bardzo trudne, gdyż ilustrują potrzebę wyrażenia preferencji, opinii, emocji, wyjaśnienia ich i odniesienia do innych osób. Krótko mówiąc, komputer współzawodniczący z człowiekiem musi być wyposażony w prawdziwą osobowość.

Już ten krótki przegląd wskazuje, że zaprogramowanie komputera o pełnej sztucznej inteligencji nie jest raczej możliwe. Są wprowadzić programy, i to jeszcze napisane w latach sześćdziesiątych, które prowadzą dialog przypominający rozmowę ludzi, ale bliższe przyjrzenie się ich działaniu natychmiast je dyskwalifikuje.

Komputer von Neumanna

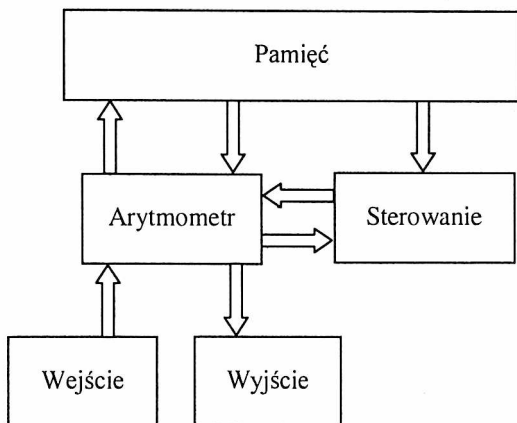
Większość współczesnych komputerów wyrasta z architektury komputerów von Neumanna. Przez pojęcie *architektura komputera* rozumie się wskazany zbiór powiązanych ze sobą elementów składowych oraz zakres ich funkcji. Pojęcie *architektura* w informatyce odnosi się także do innych elementów, np. mówimy o *architekturze programów*, mając na myśli elementy składowe (moduły, komponenty) oraz to, jak są ze sobą powiązane.

Elementami komputera o architekturze von Neumanna (rys. 2.2) są następujące bloki funkcjonalne:

- blok sterowania,
- arytmometr,
- pamięć,
- urządzenia wprowadzania danych,
- urządzenia wyprowadzania danych (wyników).

Bloki te są powiązane ze sobą magistralami komunikacyjnymi, służącymi do wymiany danych. Każdy z bloków dysponuje lub przetwarza pewne dane. Istotą pomysłu von Neumanna było umieszczenie w pamięci maszyny nie tylko danych, które mają być przetworzone, ale również algorytmu ich przetwarzania. Algorytm jest zapisany w pamięci maszyny w postaci pewnego ciągu instrukcji (poleceń), zwanego *programem*. Instrukcje są pewnym rodzajem danych. Instrukcja odpowiada pojedynczemu krokowi algorytmu, to znaczy określa pewną akcję obliczeniową oraz wskazuje kolejny krok algorytmu. Przypomnijmy, że dane są w komputerze reprezentowane w postaci ciągów binarnych, zatem zarówno dane do obliczeń, jak i program są zapamiętane w postaci ciągów zero-jedynkowych.

Rys. 2.2
Architektura komputera von Neumanna



Strzałki łączące poszczególne bloki pokazują kierunki przepływu danych. Na rysunku przedstawiono przykładowy sposób połączeń bloków. Funkcje poszczególnych bloków są następujące.

Pamięć operacyjna

Pamięć komputera składa się z tzw. komórek pamięci. Pojedyncza komórka jest pojemnikiem, który może przechowywać ciągi binarne. Składa się na ogół z ustalonej liczby bitów lub bajtów, zwykle są to 16, 32 lub 64 bity, czyli 2, 3 lub 4 bajty. Liczbę bitów lub bajtów nazywamy jej długością. Liczba komórek pamięci, nazywana pojemnością lub rozmiarem pamięci, jest zwykle potęgą liczby 2. Typowe pojemności pamięci operacyjnych współczesnych komputerów osobistych to 256 MB, 512 MB, 1 GB.

Każda komórka ma także swój unikalny adres (identyfikator). Adres jest liczbą przedstawianą w postaci binarnej. Oczywiście, w zależności od liczby komórek, czyli rozmiaru pamięci, do zapamiętania adresu jest potrzebna odpowiednia liczba bitów, np. pamięć o rozmiarze 1 M komórek, niezależnie od długości komórek, wymaga do bezpośredniej adresacji 20 bitów, gdyż $1\text{ M} = 2^{20}$.

Adres komórki oraz jej zawartość stanowią pewną daną abstrakcyjną, to znaczy, że na podstawie znajomości tylko adresu i zawartości nie można ustalić, jaka jest interpretacja tej danej. Dana ta może być interpreto-

wana jako dana właściwa (dana do obliczeń) lub jako instrukcja algorytmu, ale zależy to od ogólnych ustaleń dotyczących zawartości całej pamięci.

Pamięć będziemy oznaczać symbolem PAM, a symbolem $PAM(k)$ — k -tą komórkę pamięci, albo inaczej: komórkę pamięci o adresie k .

Na pamięci można wykonywać dwie operacje: wpisu i odczytu z wybranej komórki pamięci. Pierwsza operacja oznacza zapis do komórki $PAM(k)$ wskazanego ciągu binarnego, a jej wykonanie oznacza skasowanie poprzedniej zawartości komórki. Druga operacja oznacza odczytanie zawartości komórki $PAM(k)$, a jej wykonanie nie zmienia zawartości komórki.

W pamięci przechowuje się dane oraz programy. Zwykle obszary te są rozdzielone.

Wejście i wyjście

Bloki wejścia i wyjścia służą do komunikacji komputera z jego użytkownikiem. Wejście (odpowiednik klawiatury) pozwala na wprowadzanie danych, które są potrzebne komputerowi w czasie wykonywania programu, a wyjście (odpowiednik monitora) pozwala komputerowi na wyprowadzanie wyników obliczeń programu. Dla uproszczenia założymy, że wejście i wyjście są reprezentowane przez pojedyncze komórki pamięci zwane *rejestrami*. Rejestr bloku wejściowego będzie nazywany *wejściem*, a bloku wyjściowego — *wyjściem*. Zawartość rejestru *wejście* będzie ustalać użytkownik komputera, a rejestru *wyjście* — komputer.

Arytmometr

Arytmometr jest urządzeniem, które jest zdolne wykonywać pewien zbiór operacji arytmetycznych, np. dodawania i odejmowania liczb, oraz operacji logicznych, np. sprawdzania, czy liczba jest równa zeru.

Arytmometr ma dostęp do pamięci, z której może pobierać i do której może wpisywać dane. Dysponuje również własnymi wyróżnionymi komórkami pamięci zwanymi rejestrami. Rejestrów może być wiele, tu założymy, że są tylko cztery: jeden zwany *akumulatorem* (AK), o takiej samej długości jak komórka pamięci, oraz trzy 1-bitowe rejestry warunków o nazwach D , Z oraz N .

Akumulator jest rejestrzem, którego zawartość może być argumentem oraz wynikiem operacji wykonywanych przez arytmetr. Rejestry wa-

runków określają własności wyniku zgromadzonego w akumulatorze po wykonaniu operacji.

W przypadku rejestru D : jeżeli zawartością akumulatora jest liczba dodatnia, to jego zawartością jest 1, a jeżeli liczba niedodatnia, zawartością D jest 0.

W przypadku rejestru Z : jeżeli zawartością akumulatora jest 0, to jego zawartością jest 1, a przypadku przeciwnym — 0.

Rola rejestru N (N od nadmiar) wymaga bardziej szczegółowego wyjaśnienia. Załóżmy, że w komórkach pamięci przechowujemy tylko liczby całkowite. Za względu na ograniczoną długość komórek przechowywane liczby mogą pochodzić tylko z pewnego zakresu. Np. gdyby założyć, że pamięci przechowujemy tylko nieujemne liczby całkowite, to w komórce o długości 8 bitów można przechować liczby z zakresu od 0 do $255 = 2^8 - 1$, a komórce o długości 16 bitów — z zakresu od 0 do $65535 = 2^{16} - 1$. Wykonując pewną operację arytmetyczną, np. dodawanie, na dwóch liczbach z dopuszczalnego zakresu, możemy otrzymać wynik, który jest liczbą spoza tego zakresu — taka sytuacja jest nazywana *nadmiarem*. Nadmiar informuje, że powstała liczba nie mieści się w dopuszczalnym zakresie.

Zawartość rejestru N jest określona następująco: jeżeli podczas wykonywania operacji arytmetycznej (np. dodawania, odejmowania) nastąpił nadmiar, to jego zawartością jest 1, a w przypadku przeciwnym — 0.

Arytmometr wykonuje operacje (np. operacje arytmetyczne), które określają instrukcje. Operacje są wykonywane na zawartości komórek pamięci o wskazanych adresach oraz na akumulatorze. Efektem wykonania przez arytmmetr operacji jest również ustalenie zawartości jego rejestrów.

Repertuary operacji wykonywanych przez różne arytmmetry mogą być różne. Np. można założyć, że arytmmetr wykonuje następujące operacje:

- wpisz same zera do akumulatora,
- do zawartości akumulatora dodaj zawartość k -tej komórki pamięci,
- od zawartości akumulatora odejmij zawartość k -tej komórki pamięci,
- zawartość akumulatora zapisz w k -tej komórce pamięci,
- zawartość wejścia wpisz do akumulatora,
- zawartość akumulatora wpisz do wyjścia.

Dodatkowym efektem operacji zmieniających zawartość akumulatora jest ustalenie zawartości rejestrów warunków D , Z oraz N .

Sterowanie

Blok sterowania jest odpowiedzialny za wykonywanie programów umieszczonych w pamięci. Sterowanie wraz z arytmometrem we współczesnych komputerach stanowi zwykle całość nazywaną procesorem.

Wykonanie programu polega na cyklicznym pobieraniu i wykonywaniu kolejnych instrukcji programu, aż do zatrzymania obliczeń. Na wykonanie pojedynczej instrukcji składają się:

- realizacja określonej operacji na wskazanych argumentach,
- wskazanie do wykonania kolejnej instrukcji.

Wykonanie programu rozpoczyna się od wskazanej instrukcji. Instrukcja ta jest określona przez rejestr należący do bloku sterowania, nazywany *licznikiem rozkazów*. Jego zawartością jest adres komórki pamięci, w której znajduje się instrukcja. W stanie początkowym licznik rozkazów wskazuje na adres komórki zawierającej pierwszą instrukcję programu, a następnie — w trakcie wykonywania programu — na adres komórki zawierającej aktualnie wykonywaną instrukcję.

Instrukcje mogą być wykonywane w takiej kolejności, w jakiej są umieszczone w pamięci, ale także, w zależności od efektu wykonania instrukcji, kolejność ta może być zmieniana.

W pierwszym przypadku po wykonaniu instrukcji adres licznika zwiększa się o 1, natomiast w drugim — licznik rozkazów może zmieniać się inaczej.

Gdy licznik rozkazów wskaże na komórkę, w której znajduje się wyróżniona instrukcja *stop*, następuje zatrzymanie obliczeń programu.

Czynności wykonywane przez blok sterowania zależą od zbioru instrukcji, które może wykonywać komputer. Każda instrukcja ma nazwę, a niektóre z nich dodatkowo argument. Zestaw ten jest kompletny w tym sensie, że za pomocą instrukcji daje się przedstawić dowolny algorytm.

Zakładamy, że pojedyncza instrukcja jest reprezentowana ciągiem binarnym przechowywanym w pojedynczej komórce pamięci. W przedstawionym przykładzie liczba różnych instrukcji wynosi 10, co oznacza, że dla ich prezentacji binarnej potrzebne są 4 bity. Pozostałe bity komórki pamięci mogą być wykorzystane do zapamiętania argumentu instrukcji — adresu komórki pamięci. Zawartość komórki przechowującej instrukcję dzieli się na dwie części: część rozkazową — przechowującą kod operacji i część adresową — wskazującą na argument. Przy założeniu, że np.

długość komórki wynosi 2 bajty, 4 bity będą przeznaczone na część operacyjną i 12 bitów — na część adresową.

Tab. 2.1

Przykładowa lista instrukcji maszynowych

Lp.	Symboliczna postać	Znaczenie
1	<i>ZerujAk</i>	Wpisz 0 do akumulatora.
2	<i>DodajDoAkPam(k)</i>	Do zawartości akumulatora dodaj zawartość komórki o adresie k .
3	<i>OdejmijOdAkPam(k)</i>	Od zawartości akumulatora odejmij zawartość komórki o adresie k .
4	<i>WpiszAkDoPam(k)</i>	Zawartość akumulatora zapisz w komórce o adresie k .
5	<i>OdczytajWeDoAk</i>	Zawartość rejestru <i>wejście</i> wpisz do akumulatora.
6	<i>WpiszAkNaWy</i>	Zawartość akumulatora wpisz do rejestru <i>wyście</i> .
7	<i>JeżeliDodSkocz(k)</i>	Jeżeli zawartością rejestru D jest 1, to wykonaj instrukcję zawartą w komórce o adresie k , w przypadku przeciwnym wykonaj kolejną instrukcję.
8	<i>JeżeliZeroSkocz(k)</i>	Jeżeli zawartością rejestru Z jest 1, to wykonaj instrukcję zawartą w komórce o adresie k , w przypadku przeciwnym wykonaj kolejną instrukcję.
9	<i>JeżeliNadmiarSkocz(k)</i>	Jeżeli zawartością rejestru N jest 1, to wykonaj instrukcję zawartą w komórce o adresie k , w przypadku przeciwnym wykonaj kolejną instrukcję.
10	<i>Stop</i>	Zatrzymaj obliczenia programu.

Instrukcje w tabeli są przedstawione w postaci symbolicznej, mają nazwy, a nie binarne kody. Na początku można było posługiwać się instrukcjami wyłącznie w postaci kodów binarnych; zbiór instrukcji w takiej postaci jest językiem programowania zwanym *kodem maszynowym*. W okresie późniejszym instrukcje były przedstawiane, jak w przykładzie, w postaci symbolicznej. Ich zbiór stanowi język programowania nazywany *assemblerem*.

Przykład 1

Prosty program ma odczytać z wejścia kolejno dwie liczby, zapamiętać je, następnie dodać, a wynik dodawania umieścić w pamięci i wyprowadzić na wyjście.

Program ten jest niżej przedstawionym ciągiem instrukcji. Liczby po lewej stronie są adresami komórek pamięci, które zawierają poszczególne instrukcje, adresy są oddzielone od instrukcji dwukropkami. Teksty umieszczone po prawej stronie instrukcji, oddzielone podwójnym znakiem myślnika, stanowią komentarze wyjaśniające znaczenie instrukcji:

- 10: *OdczytajWeDoAk* -- zawartość rejestru *wejście* (pierwszą liczbę) wpisz do *akumulatora*.
- 11: *WpiszAkDoPam*(100) -- zawartość *akumulatora* (pierwszą liczbę) zapisz w komórce o adresie 100.
- 12: *OdczytajWeDoAk* -- nową zawartość rejestru *wejście* (drugą liczbę) wpisz do *akumulatora*.
- 13: *WpiszAkDoPam*(101) -- kolejną zawartość *akumulatora* (drugą liczbę) zapisz w komórce o adresie 101.
- 14: *DodajDoAkPam*(100) -- do zawartości *akumulatora* (drugą liczbę) dodaj zawartość komórki o adresie 101 (pierwsza liczba).
- 15: *WpiszAkDoPam*(102) -- zawartość *akumulatora* (sumę liczb) zapisz w komórce o adresie 102.
- 16: *WpiszAkNaWy* -- zawartość *akumulatora* (sumę liczb) wpisz do rejestru *wyjście*.
- 17: *Stop* -- zatrzymanie obliczeń.

Przed rozpoczęciem wykonywania programu ciąg ten powinien być umieszczony w pamięci. Program składa się z 8 instrukcji. Jeżeli założymy, że są one umieszczone w kolejnych komórkach pamięci o adresach od 10 do 17, to początkową zawartością licznika rozkazów będzie adres 10. Po wykonaniu programu zawartość komórek przechowujących program nie zmieni się, natomiast w komórkach o adresach 100, 101, 102 będą wpisane kolejno: pierwsza liczba, druga liczba i suma obu liczb.

Przykład 2

Jest to pewna kontynuacja przykładu poprzedniego. Założmy, że w pamięci komputera (w komórkach o adresach 100, 101) są dwie liczby. Należy zbadać, która z tych liczb jest większa, i liczbę większą wprowadzić na wyjście.

Program, umieszczony w kolejnych komórkach o adresach od 20 do 31, przedstawia się następująco:

- 20: *ZerujAk* -- wpisz 0 do *akumulatora*.
- 21: *DodajDoAkPam*(100) -- do zawartości *akumulatora* dodaj zawartość komórki o adresie 100.
- 22: *OdejmijOdAkPam*(101) -- od zawartość *akumulatora* (pierwsza liczba) odejmij zawartość komórki o adresie 101 (druga liczba).
- 23: *JeżeliDodSkocz*(28) -- jeżeli zawartością rejestru D jest 1, to kolejno wykonywaną będzie instrukcja zawarta w komórce o adresie 24, w przypadku przeciwnym — instrukcja zawarta w kolejnej komórce o adresie 28.

- 24: *ZerujAk* -- wpisz 0 do *akumulatora*.
25: *DodajDoAkPam*(101) -- do zawartości *akumulatora* dodaj zawartość komórki o adresie 101, gdyż liczba z komórki o adresie 101 jest nie mniejsza od liczby z komórki o adresie 100.
26: *WpiszAkNaWy* -- zawartość *akumulatora*, liczbę z komórki o adresie 101, wpisz do rejestru *wyjscie*.
27: *Stop* -- zatrzymanie obliczeń.
28: *ZerujAk* -- wpisz 0 do *akumulatora*.
29: *DodajDoAkPam*(100) -- do zawartości *akumulatora* dodaj zawartość komórki o adresie 100, gdyż liczba z komórki o adresie 100 jest nie mniejsza od liczby z komórki o adresie 101.
30: *WpiszAkNaWy* -- zawartość *akumulatora*, liczbę z komórki o adresie 100, wpisz do rejestru *wyjscie*.
31: *Stop* -- zatrzymanie obliczeń.

Warto zwrócić uwagę na instrukcję z komórki o adresie 23 — rozdziela ona dalsze obliczenia na jeden z dwóch fragmentów. Fragment z komórek o adresach 24-27 wykonuje się, gdy większa jest liczba z komórki o adresie 101, a w przypadku przeciwnym — fragment z komórek o adresach 28-31.

W przykładach założono, że programy oraz dane są rozmieszczone w ustalonych obszarach pamięci. Konsekwencją tego założenia jest konieczność ustalania bezwzględnych adresów argumentów instrukcji przy układaniu tekstu programu. Stosowanie założenia byłoby w praktyce niezmiernie uciążliwe, dlatego przyjmuje się zasadę adresowania względnego, która oznacza, że adresy podawane jako argumenty instrukcji nie są adresami bezwzględnymi, ale względnymi, czyli odnoszą się do początku obszaru pamięci, gdzie są przechowywane dane lub program.

Wtedy np. instrukcja z drugiego programu:

25: *DodajDoAkPam*(101)

mogłaby przyjąć postać:

25: *DodajDoAkPam*(*pd* + 1)

gdzie *pd* oznacza adres początku obszaru z danymi. Oczywiście bezpośrednio przed wykonaniem tej instrukcji musi nastąpić wyliczenie wartości *pd* + 1. Wyliczenia tego może dokonać np. blok sterowania na podstawie dostarczonych mu przed wykonaniem programu adresów początkowych obszaru danych.

Podobnie instrukcja:

23: *JeżeliDodSkocz*(28)

mogłaby przyjąć postać:

23: *JeżeliDodSkocz*(*pp* + 8)

gdzie *pp* oznacza adres początku programu.

O językach programowania

Bezpośrednie programowanie komputera w kodzie maszynowym czy też assemblerze jest bardzo uciążliwe. Dlatego powstały tzw. algorytmiczne języki programowania, które uwolniły programistę od binarnego lub symbolicznego kodowania instrukcji i od potrzeby znajomości architektury komputera, w tym rozmieszczania programu i danych w pamięci operacyjnej.

Liczba powstałych algorytmicznych języków programowania sięga tysięcy. Obecnie w powszechnym użyciu spotykamy takie języki, jak: C, C++, Java, C#, Visual Basic, Pascal, Ada, Fortran. W celu przekazania ogólnej informacji o tych językach przedstawiamy poniżej bardzo uproszczony język, określony tylko na użytek niniejszych rozważań. Prezentacja języka jest całkowicie nieformalna, polega głównie na odwoływaniu się do intuicyjnie oczywistych przykładów. Język ten należy do kategorii języków proceduralnych (inaczej: imperatywnych). Do tej samej kategorii należą również języki wymienione wyżej. Poza tym mamy jeszcze dwie inne, nie omawiane tu kategorie języków: języki funkcjonalne i języki deskryptywne (programowania logicznego).

Program wyrażony w przykładowym języku składa się z dwóch części: z części opisującej dane i części przedstawiającej instrukcje. Ogólna struktura programu wyraża się następująco:

```

program
zmienna
.....
koniecZmienna
procedura
.....
koniecProcedura
koniecProgram
```

Cały programu zawiera się pomiędzy dwoma słowami: **program** oraz **koniecProgram**, które stanowią znacznik początku i końca tekstu programu. Podobną rolę pełnią pozostałe słowa wskazujące na początek i ko-

niec wewnętrznych części programu. Występujące słowa, nazywane słowami kluczowymi języka programowania, są obowiązkowymi elementami tekstu programu. W pierwszym wykropkowanym miejscu opisuje się dane, a w drugim — instrukcje programu.

Opis danych składa się z ciągu deklaracji zmiennych — stąd nazwa początku i końca tego fragmentu. Zmienna, wyrażona przez pewną nazwę, pełni rolę symbolicznego adresu komórki pamięci.

Deklaracją pojedynczej zmiennej jest np. napis:

```
liczba: Całkowita;
```

Jest to deklaracja zmiennej o nazwie `liczba`. Druga część napisu określa tzw. typ zmiennej. Typ o nazwie `Całkowita` oznacza, że wartościami, jakie może przyjmować dana zmienna, są tylko liczby całkowite; zmienna nie może być użyta do pamiętania wartości innego typu. Średnik na końcu linii jest znacznikiem końca deklaracji zmiennej.

Ciąg deklaracji zmiennych ma przykładowo postać:

```
liczba1: Całkowita;  
liczba2: Całkowita;  
symbol: Znak;
```

Ciąg ten zawiera deklaracje trzech zmiennych: `liczba1` i `liczba2` typu `Całkowita` oraz `symbol` typu `Znak`.

Oprócz deklaracji zmiennych o wartościach skalarnych wprowadzimy deklarację zmiennych tablicowych. Przykładowa deklaracja zmiennej tablicowej o nazwie `ciąg` ma postać:

```
ciąg: tablica[10..20] Całkowita;
```

Deklaracja ta przedstawia zestaw 11 liczb całkowitych. Kolejne liczby z tego zestawu będą identyfikowane przez zmienne indeksowane:

```
ciąg[10], ciąg[11], ..., ciąg[20].
```

Zmienne te można używać w taki sam sposób, jak np. zmienne: `liczba1` i `liczba2`.

W drugiej części programu określa się ciąg instrukcji. W zbiorze instrukcji naszego języka mamy instrukcje proste i złożone.

Instrukcjami prostymi są:

- instrukcja przypisania,
- instrukcja pusta,
- instrukcja czytania urządzenia wejściowego,
- instrukcja pisanie na urządzeniu wyjściowym.

Instrukcjami złożonymi są:

- instrukcja alternatywy,
- instrukcja pętli.

Przed przedstawieniem budowy instrukcji należy wyjaśnić, czym są wyrażenia, które stanowią elementy składowe instrukcji. Wyrażenia są tekstami, których znaczeniem jest pewna wartość. Dalej będziemy wyróżniać wyrażenia tylko dwóch typów: wyrażenia liczbowe, których wartościami są liczby całkowite, oraz wyrażenia logiczne, których wartościami są wartości logiczne: *prawda* i *falsz*.

Wyrażenia liczbowe są napisami, w których mogą wystąpić liczby, zmienne typu liczbowego, symbole operacji arytmetycznych i nawiasy. Wszystkie te symbole muszą wystąpić w odpowiedniej kolejności, zgodnej z zasadami arytmetyki. Mogą być oczywiście różne typy liczbowe, ale zakładamy, że mamy do dyspozycji tylko typy całkowitoliczbowe. Przykładami wyrażen liczbowych są:

```
liczba1
ciag[12]
liczba1 + liczba2
(liczba1 - ciag[20]) x liczba2
```

Wyrażenia logiczne składają się z wyrażen liczbowych połączonych operatorami porównań, np.:

```
liczba1 < liczba2
(liczba1 - ciag[20]) = liczba2
```

Instrukcja przypisania ma postać:

zmienna = wyrażenie;

gdzie *zmienna* jest nazwą pewnej zadeklarowanej zmiennej, a *wyrażenie* jest takiego samego typu jak *zmienna*. Efektem wykonania instrukcji jest najpierw wyliczenie wartości *wyrażenia*, a następnie przypisanie wyliczonej wartości *zmiennej*, czyli obrazowo: do komórki pamięci o nazwie *zmienna* zostanie wpisana wyliczona wartość *wyrażenia*. Symbol średnika na końcu będzie zawsze oznaczał koniec instrukcji. Przykładami instrukcji są:

```
liczba1: = ciag[20] x liczba2;
ciag[12]: = ciag[12] + 1;
```

Instrukcja pusta ma postać:

```
nicNieRób;
```

Wykonanie tej instrukcji nie ma żadnych efektów, w szczególności zmienia wartości przypisanym zmiennym.

Instrukcja czytania urządzenia wejściowego ma postać:

```
czytajWejście(zmienna);
```

Oznacza ona odczytanie wartości wprowadzonej przez urządzenie wejściowe i przypisanie tej wartości do *zmiennnej*, np.:

```
czytajWejście(liczba1);
```

Instrukcja pisania na urządzenie wyjściowe ma postać:

```
piszWyjście(wyrażenie);
```

Oznacza ona obliczenie wartości *wyrażenia* i wyprowadzenie tej wartości na urządzenie wyjściowe, np.:

```
piszWyjście(liczba1 + liczba2);
```

Instrukcja alternatywy ma postać:

```
jeżeli wyrażenie  
to instrukcja1  
inaczej instrukcja2  
jeżeliKoniec;
```

Jest to instrukcja złożona — jej elementami składowymi są: *wyrażenie*, *instrukcja1* oraz *instrukcja2*. *Wyrażenie* jest dowolnym wyrażeniem typu logicznego, a *instrukcja1* oraz *instrukcja2* są dowolnymi instrukcjami lub ciągami instrukcji. Przedstawiony tekst może być pisany w jednej lub w większej ilości linii; nie ma to wpływu na znaczenie instrukcji, natomiast ma wpływ na czytelność.

Instrukcja wykonywana jest w sposób następujący: najpierw wylicza się wartość *wyrażenia*, następnie, jeżeli wyliczoną wartością jest *prawda*, wykonuje się *instrukcja1*, po czym cała instrukcja alternatywy się kończy. W przypadku przeciwnym, gdy wyliczoną wartością jest *falsz*, wykonuje się *instrukcja2*, po czym cała instrukcja się kończy.

Przykład instrukcji:

```
jeżeli liczba1 > = liczba2  
to większa: = liczba1;  
inaczej większa: = liczba2;  
jeżeliKoniec;
```

Instrukcja porównuje wartości dwóch zmiennych i zmiennej przypisuje wartość nie mniejszej z nich.

Instrukcja pętli ma postać:

```
dla parametr = początek..koniec powtarzaj  
  instrukcja  
dlaKoniec;
```

Obliczenie instrukcji pętli polega na wielokrotnym wykonaniu *instrukcji*, która jest jej elementem. *Instrukcja* oznacza tu dowolną instrukcję lub ciąg instrukcji. To, ile razy *instrukcja* będzie wykonywana, wyznacza zakres pętli określony przez wyrażenie *początek.koniec*, gdzie *początek* oraz *koniec* są wyrażeniami typu całkowitoliczbowego — takimi, że wartość wyrażenia *początek* jest nie większa od wartości wyrażenia *koniec*, czyli $\text{początek} \leq \text{koniec}$. Każdorazowe wykonanie *instrukcji* przebiega przy ustalonej wartości *parametru* pętli, inaczej: zmiennej sterującej pętli. Przy powtarzaniu obliczania *instrukcji parametr* przyjmuje kolejno wartości z zakresu pętli, czyli przy pierwszym wykonaniu *instrukcji* jego wartością jest *początek*, przy drugim — $(\text{początek} + 1)$, a przy ostatnim — *koniec*. *Parametr* może być elementem *instrukcji*.

Przykład instrukcji pętli:

```
dla i = 10..15 powtarzaj  
suma: = suma + ciag[i];  
dlaKoniec;
```

Instrukcja do wartości zmiennej *suma* dodaje sumę wartości zmiennej tablicowej *ciag* z zakresu od 10 do 15.

Poniżej przedstawione są dwa proste przykłady programów zapisanych w omówionym języku programowania.

Program 1

Pierwszy program odczytuje z wejścia ciąg liczb całkowitych, znajduje liczbę największą i wyprowadza ją na wyjście. Zakłada się, że dane, które użytkownik podaje kolejno na wejściu, to długość ciągu, a następnie kolejne elementy ciągu.

```
program  
zmienna  
bieżąca: Całkowita;  
największa: Całkowita;  
długośćCiagu: Całkowita;  
procedura  
czytajWejście(długośćCiagu);  
czytajWejście(największa);  
dla i = 2..długośćCiagu powtarzaj  
czytajWejście(bieżąca);  
jeżeli największa < bieżąca  
to największa: = bieżąca;  
inaczej nicNieRób;  
jeżeliKoniec;
```

```
dlaKoniec;  
piszWyjście(największa);  
proceduraKoniec;  
programKoniec;
```

Złożoność algorytmu, który reprezentuje program, jest proporcjonalna do liczby n określającej długość ciągu, co zapisujemy w postaci $O(n)$.

Program 2

Jest to program, który sortuje tablicę liczb. Zakłada się, że dane, które użytkownik podaje kolejno na wejściu, to długość ciągu, a następnie kolejne elementy ciągu. Długość ciągu musi zawierać się w przedziale od 2 do 100. Program wczytuje dane wejściowe, a następnie sortuje wprowadzony ciąg od najmniejszej do największej i wyprowadza na wyjście. Wczytane elementy ciągu będą zapisywane w zmiennej tablicowej o nazwie ciąg. Przedstawiony program realizuje tzw. algorytm sortowania bąbelkowego.

```
program  
zmienna  
ciąg: tablica[1..100] Całkowita  
większa: Całkowita;  
długośćCiagu: Całkowita;  
procedura  
czytajWejście(długośćCiagu);  
dla i = 1..długośćCiagu powtarzaj  
czytajWejście(ciąg[i]);  
dlaKoniec;  
dla j = 1..długośćCiagu - 1 powtarzaj  
dla i = 1..długośćCiagu - 1 powtarzaj  
jeżeli ciąg[i] < ciąg[i + 1]  
to nicNieRób  
inaczej większa: = ciąg[i];  
ciąg[i]: = ciąg[i + 1];  
ciąg[i + 1]: = większa;  
jeżeliKoniec;  
dlaKoniec;  
dlaKoniec;  
dla i = 1..długośćCiagu powtarzaj  
piszWyjście(ciąg[i]);  
dlaKoniec;  
proceduraKoniec;  
programKoniec;
```

Program przed wykonaniem jest umieszczany w pamięci komputera. Tam też mogą się znajdować dane wejściowe i wyjściowe. Obszary pamięci, w których znajdują się dane (w tym także programy, gdyż można uważać je za szczególny rodzaj danych), są jednoznacznie identyfikowane i nazywane plikami.

Złożoność algorytmu, który reprezentuje program, jest proporcjonalna do kwadratu liczby n określającej długość ciągu, co zapisujemy w postaci $O(n^2)$. Warto zaznaczyć, że są inne algorytmy o mniejszej złożoności obliczeniowej, np. $O(n \log(n))$. Może wydawać się, że różnica złożoności obliczeniowej nie jest duża, ale jeżeli za n weźmiemy duże liczby, np. 10^5 (co wcale nie jest rzadkie), łatwo przekonać się o różnicy liczby operacji dla takich algorytmów.

Translatory

Wyróżnia się trzy poziomy języków programowania: najniższy poziom reprezentują języki maszynowe (autokody), wyższy — języki symboliczne (asemblery), a najwyższy — języki algorytmiczne.

Języki algorytmiczne służą do takiego zapisu algorytmu, który z jednej strony jest wygodny dla programisty, a z drugiej — jest rozumiany przez komputer, to znaczy komputer na podstawie zapisu algorytmu jest w stanie przeprowadzić obliczenia algorytmu.

Programowanie w języku maszynowym jest niezwykle uciążliwe, gdyż wymaga od programisty szczegółowej znajomości architektury komputera, w tym pamiętania binarnych kodów instrukcji. Mniej uciążliwe jest pisanie w assemblerze, gdyż zwalnia programistę przynajmniej od pamiętania kodów instrukcji, ale nadal wymaga znajomości architektury komputera.

Natomiast pisząc program w języku algorytmicznym, programista jest uwolniony od konieczności jakiegokolwiek znajomości architektury komputera. Nasuwa się oczywisty wniosek, że do programowania należy używać języków algorytmicznych. Postępowanie takie jest możliwe, lecz trzeba zdawać sobie sprawę, że program napisany w assemblerze lub w języku algorytmicznym nie może być bezpośrednio wykonywany przez komputer. Komputer może wykonywać tylko programy zapisane w języku maszynowym. Oznacza to, że program napisany w języku wyższego poziomu, zanim zostanie przekazany komputerowi do wykonania, musi być poddany transformacji, która polega na zastąpieniu jego instrukcji na instrukcje zapisane w języku maszynowym. Transformację programu na-

pisanego w języku wyższego poziomu na program w języku maszynowym nazywamy *translacją programu*, albo krótko: *translacją*.

Nasuwa się pytanie: jak dokonywać translacji? Transformacja dowolnego programu w pewnym języku na odpowiadający mu program w innym języku jest przykładem problemu, dla którego daje się ułożyć algorytm. Algorytm ten można zapisać w dowolnym języku, w szczególności w języku maszynowym. Jeżeli jest zapisany w języku maszynowym, to znaczy, że jest programem, który może być wykonywany przez komputer. Program taki nazywa się *translatorem*.

Translator jest programem, który przetwarza programy, bowiem za daną wejściową przyjmuje dowolny program P_{JP} (plik wejściowy, w którym jest umieszczony program źródłowy), napisany w ustalonym języku programowania JP , a wynikiem jego obliczeń — daną wyjściową jest program P_{JM} (plik wyjściowy, w którym jest umieszczony program wykonywalny), napisany w języku maszynowym JM . Program P_{JP} jest równoważny programowi P_{JM} , co oznacza, że dla dowolnej danej wejściowej d_{we} jeżeli program P_{JP} kończy się i produkuje daną wyjściową d_{wy} , to dla tej samej danej wejściowej d_{we} program P_{JM} również kończy się i produkuje taką samą daną wyjściową d_{wy} .

Zatem wykonanie programu P_{JP} napisanego w języku wyższego poziomu JP przebiega w dwóch etapach: w pierwszym etapie za pomocą translatora dokonuje się translacji programu P_{JP} na równoważny mu program wykonywalny P_{JM} w języku maszynowym JM , a w drugim — następuje wykonanie programu P_{JM} , czyli na podstawie danych wejściowych d_{we} oblicza się dane wyjściowe d_{wy} .

Rys. 2.3

Schemat (a) translacji, (b) wykonania programu

$$\begin{array}{ll} (a) & P_{JP} \xrightarrow{\text{Translator}} P_{JM} \\ (b) & d_{we} \xrightarrow{P_{JM}} d_{wy} \end{array}$$

Translatory różnych języków programowania są stałym elementem oprogramowania komputera, a także głównym elementem *środowisk programistycznych*, tj. zestawu programów, które służą do pisania i uruchamiania programów. Innymi elementami środowiska programistycznego są m.in.:

- edytory — programy służące do redakcji tekstu programów,
- analizatory składniowe – programy służące do badania poprawności powstałych tekstów programów, chodzi np. o sprawdzenie, czy właściwie są rozmieszczone słowa kluczowe, czy zmienne występujące w instrukcjach mają swoje deklaracje;
- konsolidatory — programy, które do napisanego programu dołączają inne programy, tzw. podprogramy biblioteczne. Chodzi tu o to, że w powstających programach często pewne fragmenty — podprogramy — mogą się powtarzać, więc zamiast powtarzać ich pisanie w każdym nowym programie, dopuszcza się umieszczenie w programie tylko odwołań do podprogramów. Konsolidator rozpoznaje w programie takie odwołania i na ich miejsce wstawia wcześniej napisane i zgromadzone w bibliotece podprogramy.

Napisanie poprawnego programu nie jest zadaniem prostym. Po pierwsze, program musi być poprawny składniowo. Oznacza to, że tekst programu, podobnie jak wypowiedź w dowolnym języku, musi być zgodny z ustalonymi regułami gramatycznymi. Program poprawny składniowo nie oznacza jeszcze poprawności semantycznej, podobnie jak poprawne składniowo zdanie w języku polskim może nie mieć żadnego sensu, np. zdanie: *Mężczyzna stoi szybko*.

Program musi być poprawny semantycznie, co oznacza, że jeżeli rozpocznie swoje obliczenie dla ustalonych danych początkowych, to po zakończeniu dostarczy oczekiwanych danych końcowych, np. program, który ma obliczać drugą potęgę liczby naturalnej, powinien dla liczby początkowej 4 dostarczyć jako wyniku liczby 16.

Sprawdzenie, że napisany program jest semantycznie poprawny, odbywa się najczęściej przez testowanie, czyli porównanie wyników obliczeń programu, dla danego zestawu danych początkowych, z wynikami przewidywanymi. Proces sprawdzania poprawności składniowej i semantycznej programu nazywa się też *uruchamianiem programu*.

III. PODSTAWOWE INFORMACJE O KOMPUTERZE OSOBISTYM

Rodzaje komputerów

Różnorodność naszych komputerów jest ogromna. Różnią się one od siebie wielkością, wydajnością lub pełnioną rolą. Mogą pracować autonomicznie lub, co staje się regułą, w sieci komputerowej. Spotykamy je w biurach, zakładach pracy, a także w najrozmaitszych urządzeniach, także domowych.

Coraz częstsze są sytuacje, gdy trudno jednoznacznie stwierdzić, czy mamy do czynienia z komputerem, czy z innym rodzajem urządzenia. Jak np. traktować rozbudowany telefon komórkowy, który poza usługami telekomunikacyjnymi oferuje zestaw usług obliczeniowych, i jak odróżniać go od komputera osobistego wyposażonego w mechanizm nawiązywania i utrzymywania połączeń telekomunikacyjnych?

Mamy współcześnie do czynienia z dwiema skojarzonymi tendencjami rozwoju techniki komputerowej. Z jednej strony pojawia się coraz większa różnorodność urządzeń, w których komputer jest ważną częścią składową, a z drugiej — następuje integracja usług, które były oferowane dotąd niezależnie od siebie. Najbardziej istotną jest integracja usług obliczeniowych, telekomunikacyjnych i telewizyjnych. Czas, w którym komputer domowy będzie pełnić jednocześnie role tradycyjnego komputera, telefonu i telewizora, wydaje się niezbyt odległy. Z punktu widzenia techniki jest to możliwe już od kilku lat, na drodze do pełnej integracji stoją problemy ekonomiczne, prawne i organizacyjne (standardy), dotyczące zwłaszcza telewizji cyfrowej.

Najczęściej mamy do czynienia z komputerami osobistymi. W zasadzie są one przeznaczone do użytku przez pojedynczych użytkowników, mogą pracować autonomicznie lub w sieci komputerowej. Wśród nich można spotkać:

- komputery biurkowe (*desktop*),
- komputery przenośne (*laptop, notebook*)
- komputery podręczne (*hand held, palmtop*).

Ostatnio pojawił się nowy rodzaj komputera nazywany tabletem, który jest pośrednią formą między komputerem przenośnym a podręcznym. Wyglądem przypomina on podkładkę do pisania formatu A4, po której można pisać rysikiem, a rolę klawiatury pełni ekran dotykowy.

Wymienione rodzaje komputerów różnią się przede wszystkim wielkością i wydajnością, w mniejszym stopniu zakresem oferowanych funkcji, co jest oczywiste, gdyż możliwości komputera o większych gabarytach, przy założeniu tej samej technologii wykonania, są większe od komputera mniejszego.

Większą wydajnością niż komputery osobiste charakteryzują się komputery przeznaczone do obsługi wielu użytkowników. Obecnie są one nazywane *serwerami*, a użytkownicy, dysponując komputerami osobistymi (stacjami roboczymi), korzystają z jego usług za pośrednictwem sieci komputerowej. Dawniej takie komputery nazywano *komputerami głównymi (mainframe)*, a użytkownicy korzystali z ich usług za pomocą bezpośrednio podłączonych do nich indywidualnych monitorów.

Wśród serwerów jest także różnorodność rozwiązań. Najprostszą formą jest pojedynczy, rozbudowany komputer, bardziej rozbudowaną — *klaster* lub *farma komputerów*, a najbardziej rozbudowaną — *superkomputer*.

Farma komputerów składa się z wielu, zwykle identycznych, ulokowanych w jednym miejscu komputerów. Z punktu widzenia użytkownika jest ona widziana jako jeden komputer, gdyż użytkownik nie ma możliwości odróżnienia od siebie poszczególnych komputerów farmy. Zlecenia usług kierowane przez użytkownika trafiają do jednego z komputerów farmy — *rozdzielacza*, który decyduje o przekazaniu zlecenia do innego komputera farmy.

Superkomputery są komputerami o bardzo dużej wydajności, którą osiąga się przez bezpośrednie połączenie wielu (obecnie nawet kilku tysięcy) procesorów w jeden komputer. Zasadnicza idea przyspieszająca konstrukcji superkomputera wynika z tego, że przyspieszenie obliczeń programu można osiągnąć przez podzielenie programu na części składowe i wykonywanie tych składowych jednocześnie na różnych procesorach. Wykorzystanie możliwości superkomputera wymaga opracowania algorytmu obliczeń, który przewiduje możliwość równoległego wykona-

nia przynajmniej części obliczeń. Algorytmy takie nazywamy *algorytmami równoległymi*.

Wydajność komputerów mierzy się w różny sposób. Sama częstotliwość pracy zegara komputera nie rozstrzyga o szybkości przetwarzania, która zależy w równej mierze od sprzętu, jak i oprogramowania. Wydajność komputera określa się przez czas potrzebny na obliczenie pewnej mieszanki programów. Inną miarą wydajności jest liczba operacji arytmetycznych wykonywanych w jednostce czasu, jednostką wydajności jest liczba tzw. operacji zmiennoprzecinkowych (operacji na liczbach rzeczywistych) w ciągu jednej sekundy — FLOPS (*Flow Operations Per Second*). Najwyższa wydajność osiągana przez współczesne najszybsze superkomputery sięga 10^{12} FLOPS.

Komputery osobiste występują w zróżnicowanych konfiguracjach sprzętowych i programowych.

Elementy konfiguracji sprzętowej

Elementami pierwszego komputera o architekturze von Neumanna są: blok sterowania i arytmometr — stanowiące procesor, pamięć operacyjna, bloki wprowadzania i wyprowadzania danych. Każdy z tych elementów ma odpowiedniki w architekturze współczesnego komputera, przy czym są one bardziej rozbudowane i zróżnicowane.

Patrząc na komputer osobisty stojący na biurku, zauważamy co najmniej trzy połączone podzespoły: jednostkę centralną, klawiaturę i monitor ekranowy.

W jednostce centralnej, o różnych wielkościach obudowy, mieszczą się elementy elektroniczne wraz z zasilaniem elektrycznym i wentylacją. Elementy elektroniczne mają budowę modułową. Większość z nich — procesor, pamięć operacyjna, sterowniki urządzeń zewnętrznych — jest umieszczona na tzw. płycie głównej. Poza płytą główną są umieszczone pamięci masowe oraz sterowniki innych urządzeń.

Inne urządzenia zewnętrzne współpracujące z komputerem to m.in.:

- pamięci zewnętrzne: dyskiety i taśmy magnetyczne, dyski optyczne,
- urządzenia wejściowe: mysz, mikrofon, kamera, skaner,
- urządzenia wyjściowe: drukarki, głośniki.

Coraz częściej komputer jest wyposażony w urządzenia umożliwiające dostęp do Internetu za pośrednictwem sieci telefonicznej, telewizyjnej lub lokalnych sieci komputerowych.

Poniżej prezentujemy krótki przegląd i ogólną charakterystykę elementów wchodzących w skład konfiguracji komputera osobistego.

Płyta główna

Płyta główna zawiera przede wszystkim procesor — najważniejszy element komputera, pamięć operacyjną oraz tzw. karty sterujące — urządzenia umożliwiające współpracę procesora z monitorem, klawiaturą i innymi urządzeniami. Karty sterujące są w istocie wyspecjalizowanymi procesorami, przeznaczonymi do wykonywania stałych funkcji, np. karta muzyczna, karta graficznej obsługi monitora, modem lub karta sieciowa. Elementy umieszczone na płycie głównej wymieniają ze sobą dane za pomocą magistrali komunikacyjnej. Magistrale są zbudowane w oparciu o ustalone standardy, które określają m.in. częstotliwość pracy procesora, prędkość transmisji danych oraz zestaw możliwych urządzeń współpracujących. Standardy dotyczą także portów (złączy wraz z ustalonym sposobem wymiany danych) i sterowników (urządzeń sterujących) podłączanych urządzeń.

Rozróżnia się kilka rodzajów portów. Podstawowy podział dzieli porty na szeregowy (wolniejszy) i równoległy (szybszy). W portach szeregowych transmisja danych przebiega sekwencyjnie, bit po bicie, podczas gdy w portach równoległych transmisja przebiega znakami (bajtami) — wszystkie bity znaku są przekazywane jednocześnie przez kilka linii (przewodów). Porty szeregowy są wykorzystywane np. do komunikacji z myszką lub modemem, a równoległy — do komunikacji z drukarką, skanerem i zewnętrznymi napędami dysków.

Na płycie głównej są także umieszczane procesory pomocnicze i pamięci stałe zawierające programy służące do organizacji pracy komputera. Programami takim są np. BIOS (*Basic Input Output System*), który m.in. rozpoznaje rodzaje dołączanych urządzeń, takich jak: czytniki dyskietek lub dysków oraz CHIPSET, który odpowiada za komunikację poszczególnych komponentów umieszczonych na płycie głównej z procesorem.

Procesor

Procesor jest najważniejszym składnikiem komputera. Ze względu na niezwykle wysokie wymagania związane z produkcją, a zwłaszcza rozwojem procesorów opłacalna jest tylko produkcja masowa na skalę światową.

wą. W związku z tym liczba producentów procesorów maleje, obecnie rynek komputerów osobistych zdominowały dwie firmy: Intel i AMD.

Z punktu widzenia użytkownika komputera pośrednią miarą wydajności procesora są dwa parametry: częstotliwość jego zegara synchronizującego oraz rozmiar jego wewnętrznej pamięci notatnikowej.

Zegar wyznacza podział czasu na odcinki, podczas których poszczególne elementy procesora wykonują przydzielone im czynności. Obecnie typowa częstotliwość zegara synchronizującego wynosi pojedyncze GHz (gigaherce).

Natomiast rozmiar pamięci notatnikowej (*cash*) wynosi ok. 512 KB. Pamięć wewnętrzna jest bardzo szybka, ale jednocześnie bardzo droga (około połowy ceny procesora). Miarą szybkości pamięci jest czas dostępu do pamięci — czas, jaki upływa od momentu, w którym procesor żąda dostępu do zawartości komórki, do momentu uzyskania tej zawartości. W przypadku pamięci notatnikowej okres ten wynosi ułamki nanosekund ($1 \text{ ns} = 10^{-9} \text{ s}$), podczas gdy czas dostępu do pamięci operacyjnej wynosi pojedyncze nanosekundy, a dostęp do dyskowej pamięci magnetycznej wynosi kilka, kilkanaście milisekund ($1 \text{ ms} = 10^{-3} \text{ s}$).

Warto przypomnieć tzw. prawo Moore'a — byłego prezesa Intela, który przed prawie 40 laty powiedział półzartem, że co 18 miesięcy wydajność procesorów się podwaja, a ich ceny spadają o połowę. Prawo to zadziwia swoją trafnością do dziś. Według aktualnych prognoz taki trend może trwać do ok. 2020 r., kiedy nastąpi kres rozwoju obecnie wykorzystywanej technologii półprzewodnikowej. Początki tej technologii sięgają końca lat pięćdziesiątych ubiegłego wieku. Najpierw wykorzystywano pojedyncze tranzystory do konstrukcji układów cyfrowych, później rozpoczął się proces integracji zestawów tranzystorowych w coraz bardziej złożone układy funkcjonalne. Obecnie tzw. układy wielkoscalone stanowią układ warstw materiału półprzewodnikowego i przewodnika o grubościach tysięcznej części ludzkiego włosa. Dalsze zmniejszanie grubości do warstw o grubości kilku atomów doprowadzi do pojawienia się zjawisk fizycznych, które uniemożliwią funkcjonowanie układów w oparciu o dotychczas przyjmowane zasady (jeden z kierunków prowadzonych aktualnie badań wiąże się z tzw. komputerami kwantowymi).

Rozwój technologii elektronicznej doprowadził do tego, że coraz częściej stawia się pytania o ograniczenia fizyczne. Poza omówionym ograniczeniem są jeszcze dwa inne: rozchodzenie się sygnałów elektrycznych z prędkością światła oraz odprowadzanie ciepła wydzielającego się w układzie w wyniku przełączania.

Z pierwszego ograniczenia wynika zależność pomiędzy maksymalnym wymiarem liniowym L procesora a częstotliwością pracy zegara ω , mianowicie: $\omega < c/L$, gdzie c oznacza prędkość światła. Chodzi o to, że wymiar liniowy nie może przekroczyć długości potrzebnej na rozejście się jednego impulsu synchronizującego. Ze wzoru wynika, że np. przy zegarze 2 GHz maksymalny rozmiar liniowy procesora nie może przekraczać 15 cm.

Drugie ograniczenie trzeba brać pod uwagę przy określeniu maksymalnej szybkości przełączania układów. Każdy procesor, który przełącza się z częstotliwością większą niż 10^{15} Hz, ulegnie zniszczeniu cieplnemu, niezależnie od pomysłowości konstruktorów. Przypomnijmy, że obecnie stosowane częstotliwości wynoszą pojedyncze GHz, a $1 \text{ GHz} = 10^9 \text{ Hz}$.

Pamięci

Pamięci współczesnych komputerów nie są jednorodne. Komputer dysponuje kilkoma rodzajami pamięci, które różnią się istotnymi parametrami. Zasadniczymi parametrami, które zwykle bierze się pod uwagę, są:

- szybkość pamięci — czas dostępu i prędkość transmisji,
- jednostkowy koszt (np. 1 bajtu),
- trwałość zapisu.

Szybkość pamięci jest ściśle skojarzona z kosztem jednostkowym — koszt jednostkowy rośnie gwałtownie ze wzrostem szybkości pamięci. Powoduje to, że projektanci wprowadzają kilka poziomów hierarchii pamięci. Pamięci są umieszczane wewnątrz i na zewnątrz komputera. Typowe rozwiązania obejmują:

- superszybką, względnie małą, poniżej 1 MB, pamięć notatnikową (podręczną) wewnątrz procesora,
- szybką pamięć operacyjną, nazywaną też pamięcią RAM (*Random Access Memory*), o rozmiarach poniżej 1 GB,
- wolną pamięć wewnętrzną na dyskach magnetycznych, o rozmiarach dziesiątków GB,
- wolną i małą (1,4 MB) pamięć zewnętrzną na dyskietkach magnetycznych,
- wolną i dużą (kilkanaście GB) pamięć na dyskach optycznych CD (*Compact Disc*) i DVD (*Digital Versatile Disc*, poprzednio *Digital Video Disc*),

– duże zewnętrzne pamięci dyskowe lub taśmowe o pojemnościach mierzonych w TB dla serwerów przechowujących duże bazy danych.

Rozmiary poszczególnych rodzajów pamięci należy dobierać stosownie do przewidywanych zastosowań komputera. Mechaniczne zwiększanie rozmiaru pamięci, nawet tej najszybszej, może nie wpłynąć na wydajność komputera podczas wykonywania określonych aplikacji.

Pamięci różnią się trwałością zapisu — wyróżnia się pamięci trwałe i tymczasowe. Pamięć jest trwała, jeżeli jej zawartość nie ulega zmianie po wyłączeniu zasilania, natomiast pamięć tymczasowa zachowuje swą zawartość tylko podczas zasilania komputera. Pamięci notatnikowa i operacyjna są pamięciami tymczasowymi, natomiast pamięci magnetyczne — dyskowa i taśmowa — są pamięciami trwałymi.

Pamięć notatnikowa i operacyjna są pamięciami półprzewodnikowymi, co nie oznacza jednak, że wszystkie pamięci półprzewodnikowe są tymczasowe. Wśród pamięci półprzewodnikowych są również pamięci trwałe, ale ograniczone tym, że ich zawartość nie może ulegać zmianie — nazywamy je pamięciami stałymi lub pamięciami ROM (*Read-Only Memory*). Służą one do pamiętania programów stałych umieszczanych na płycie głównej, takich jak BIOS czy CHIPSET.

Nowym rodzajem półprzewodnikowej pamięci stałej jest pamięć typu *flash*. Ma ona ograniczenie polegające na tym, że maksymalna liczba operacji zapisu do pamięci jest rzędu kilku tysięcy. Pamięć jest szybka, jej wielkość wynosi kilkaset MB; używana jest powszechnie w cyfrowych aparatach fotograficznych.

Podobnie trwałe i stałe są pamięci na dyskach optycznych, przy czym nowsze rozwiązania pozwalają na zapis wielokrotny. Dyski CD są wypierane powoli przez dyski DVD.

Klawiatura

Klawiatura jest podstawowym urządzeniem wejściowym, za pomocą którego użytkownik może wprowadzać do komputera dane w postaci tekstowej.

Ze względu na rozwiązania fizyczne wyróżniamy klawiatury dotykowe (sensorowe) i mechaniczne oraz klawiatury przewodowe i bezprzewodowe.

Funkcję klawiatury może też pełnić ekran dotykowy. Takie rozwiązanie jest często spotykane w przypadku ekranów monitorów służących do przekazywania informacji na lotniskach, w hipermarketach, urzędach, ban-

kach, a także w komputerach podręcznych. W tych ostatnich coraz częściej występują rozwiązania, które pozwalają na ręczne pisanie rysikiem na ekranie. Tak wprowadzony tekst jest automatycznie rozpoznawany i zamieniany na pismo drukowane, na którym można dalej pracować jak na zwykłym tekście wprowadzonym za pomocą klasycznej klawiatury. Okazuje się, że powszechnie obowiązujący model klawiatura – monitor nie jest niezastąpiony.

Klawiatury różnią się ponadto zestawem klawiszy i kształtem. Zwykle możemy wyróżnić cztery grupy klawiszy:

- klawisze alfanumeryczne, służące wprowadzaniu liter, cyfr i znaków interpunkcyjnych (układ przeniesiony z tradycyjnych mechanicznych maszyn do pisania),
- klawisze numeryczne (układ przeniesiony z kalkulatorów elektronicznych),
- klawisze sterowania kursorem,
- różne zestawy klawiszy funkcyjnych.

Producenci starają się nadać klawiaturze kształt ergonomiczny, ułatwiający szybkie pisanie. Do tradycyjnego, powszechnie używanego kształtu zaproponowano podkładkę pod nadgarstki, która ułatwia trzymanie rąk nad klawiaturą. Dalej posuniętym rozwiązaniem jest klawiatura „łamana”, składająca się z dwóch, ustawionych pod różnym kątem części: dla lewej i prawej ręki.

Przyciśnięcie klawisza lub kombinacji klawiszy oznacza wprowadzenie do komputera pewnego znaku. Znak jest w pamięci komputera reprezentowany pewnym ciągiem binarnym — kodem znaku. Repertuar wprowadzanych znaków i ich kodów zależy od przyjętego standardu. Jednym z najpowszechniej stosowanym jest wspomniany standard ASCII, używany do reprezentowania znaków alfanumerycznych i sterujących.

Mysz

Obok klawiatury nieodłącznym uzupełniającym urządzeniem wejściowym stała się mysz. W porównaniu do klawiatury mysz nie daje wprowadzić nowych możliwości wprowadzania danych, ale znacznie usprawnia i przyspiesza wykonywanie niektórych czynności. Mysz umożliwia sterowanie obiektami widocznymi na ekranie monitora i zastępuje klawisze klawiatury przeznaczone do sterowania kursorem ekranu i akceptacji poleceń.

Podobnie jak w przypadku klawiatury, mamy różnorodność rozwiązań fizycznych — wyróżniamy myszy mechaniczne i optyczne oraz myszy

przewodowe i bezprzewodowe, te ostatnie w dwóch wersjach: sterowane podczerwienią (rozwiązanie starsze) i falami radiowymi (rozwiązanie nowsze).

Tradycyjna mysz posiada dwa lub trzy przyciski zastępujące klawisze klawiatury *Enter* lub *Esc*, w zależności od zdefiniowanej opcji, oraz sensor (mechaniczna kulka lub element optyczny) rejestrujący zmianę położenia myszy na podkładce. Obecnie produkowane myszy są wzbogacone o dodatkowe przyciski, pełniące funkcję lupy (powiększania wskazanego obszaru ekranu) oraz pokrętła do pionowego i poziomego przesuwania obrazu.

Odmianą myszy jest kot (*trackball*). Jest to odwrócona mysz — zamiast obracać kulką przez przemieszczanie myszy, obracamy kulką bezpośrednio palcami, sterując w ten sposób kursorem. Takie rozwiązania spotykamy zwykle w komputerach przenośnych. Za szczególnie rodzaj kota lub myszy, jak kto woli, można uważać dotykową podkładkę na komputerach przenośnych, gdzie kursorem steruje się przez przesuwanie palcem po podkładce.

Za jeszcze inną odmianę myszy można uznać *joystick*, który jest wygodnym manipulatorem w grach komputerowych.

Monitor

Monitory ekranowe są podstawowym urządzeniem wyjściowym, na których wyświetla się teksty i obrazy.

Ze względu na konstrukcję wyróżnia się monitory kineskopowe — CRT (*Cathode Ray Tube*), ciekłokrystaliczne — LCD (*Liquid Crystal Display*) i plazmowe. Dwa ostatnie pozwalają na płaską konstrukcję ekranu. Zapowiadane są inne konstrukcje, m.in. elastyczne ekrany polimerowe, które mają dać się zwijać w rulonik lub nawet składać jak kartka papieru.

Obecnie nie spotyka się już monitorów monochromatycznych, wszystkie monitory wyświetlają obraz kolorowy. Różni je liczba wyświetlanych kolorów, choć i tutaj kształtuje się pewien standard wymagań minimalnych — ponad 16 tys. kolorów.

Zasadniczymi parametrami, ważnymi z punktu widzenia użytkownika, są wielkość i rozdzielczość ekranu.

Wielkość ekranu mierzy się długością przekątnej wyrażonej w calach. Zależy to od rodzaju wykonywanej pracy, np. duży ekran jest niezbędny w przypadku pracy poligraficznej czy przy projektowaniu grafiki. Spoty-

kane wielkości ekranów zależą od typu monitora, zazwyczaj ekrany kineskopowe mają wielkość 13-38 cali, ciekłokrystaliczne — 15-24 cale, a plazmowe — 34-61 cali.

Obraz na ekranie składa się z małych jednobarwnych punkcików zwanych pikselami. Kolor i jasność każdego z pikseli są sterowane oddzielnie. Ekran jest podzielony na piksele siatką linii poziomych i pionowych. Liczba tych linii wyznacza rozdzielczość ekranu. Typową, powszechnie spotykaną jest rozdzielczość 1024×768 .

Istotnym parametrem monitorów kineskopowych jest częstotliwość odświeżania obrazu, powinna ona wynosić co najmniej 75 Hz — wówczas nie jest odczuwalny efekt mrugania obrazu, oraz wielkość plamki omiatającej ekran — od 0,20 do 0,28 mm; im mniejsza plamka, tym bardziej wyraźny jest obraz.

Pełne wykorzystanie możliwości danego monitora jest uwarunkowane jakością karty graficznej obsługi monitora, która jest zainstalowana w jednostce centralnej komputera. Karta graficzna (karta video, adapter graficzny) jest w istocie wyspecjalizowanym procesorem z własną pamięcią wewnętrzną, która służy do zapamiętywania wyświetlanego obrazu; jej wielkość powinna być większa od 16 MB. Kartę charakteryzują te same parametry co monitor: liczba kolorów, rozdzielczość, częstotliwość odświeżania. Szczególnie istotny jest dobór karty, gdy oczekujemy wyświetlania nie tylko grafiki, ale i obrazów ruchomych. W rozwiązaniach specjalnych karta graficzna jest wyposażona w tzw. akcelerator graficzny, który przyspiesza rysowanie obrazów.

Drukarki

Drukarka jest uzupełniającym urządzeniem wyjściowym. Używane są głównie drukarki mozaikowe, laserowe i atramentowe.

Drukarki mozaikowe (igłowe) są rozwiązaniem najstarszym, choć nadal stosowanym, szczególnie tam, gdzie jest wymagany wydruk dokumentów w kilku kopiach. Podstawą ich różnicowania jest liczba igieł używanych do wydruku — mamy drukarki 9-, 12- i 24-igłowe (większa liczba igieł podnosi jakość wydruku). Drukarki te dobrze radzą sobie z wydrukiem w trybie znakowym, ale wydruk w trybie graficznym pozostawia wiele do życzenia, nie pozwalają natomiast na wydruk obrazów kolorowych.

Drukarki laserowe i atramentowe pozwalają na uzyskanie wydruku wysokiej jakości obrazów kolorowych. Drukarki laserowe są droższe od

atramentowych, ale za to tańsze w eksploatacji. W praktyce w biurach, zakładach pracy przeważają drukarki laserowe, gdyż są wydajne, wytrzymują duże obciążenie ilością wydruków i dają trwały efekt (nie blaknie i nie ściera się), natomiast użytkownicy domowi używają raczej drukarek atramentowych.

Istotnym elementem kosztów eksploatacji drukarki atramentowej jest liczba kałamarzy. Najprostsze są drukarki jednokałamarzowe, ale podczas eksploatacji zmuszeni jesteśmy zmieniać kałamarz z atramentem czarnym na kolorowy, gdy napis lub rysunek musi być kolorowy. Najbardziej popularne są obecnie drukarki dwukałamarzowe (jeden pojemnik z tuszem czarnym i drugi z kolorowym), a od pewnego czasu pojawiły się drukarki czterokałamarzowe — każdy z podstawowych kolorów jest w oddzielnym pojemniku i wyczerpanie jednego z kolorów podstawowych ogranicza się do wymiany jednego pojemnika.

Istotną cechą drukarek jest rozdzielczość w poziomie i pionie podawana w liczbie punktów na cal, w skrócie DPI (*Dots Per Inch*). Wśród producentów można zauważyć dwa podstawowe systemy rozdzielczości: wielokrotność 360 lub 600 DPI. Jako minimum można przyjąć 600×600 lub 720×720 DPI.

Obecnie większość drukarek potrafi drukować na kopertach, na folii oraz na papierze o różnych grubościach. Typowa szybkość wydruku tekstu to kilka, kilkanaście stron na minutę, drukowanie obrazów jest na ogół dłuższe i zależy od zadanej rozdzielczości. Standardowym formatem papieru jest A4, drukarki o większych formatach A3, A2 są znacznie droższe.

Skaner

Skaner jest urządzeniem wejściowym czytającym dokumenty lub obrazy zapisane na papierze. Obraz wejściowego dokumentu jest poddany dyskretyzacji, to znaczy jego powierzchnia jest podzielona na piksele. Wczytany dokument, niezależnie od tego, czy zawiera tekst, czy grafikę, jest zapamiętany jako obraz.

Wyróżnia się skanery monochromatyczne i kolorowe. Podobnie jak w przypadku drukarek, istotna jest ich rozdzielczość. Rozdzielczość decyduje o rozpoznawaniu szczegółów, np. przy rozpoznawaniu małych polskich liter zalecana jest minimalna rozdzielczość 1200 DPI.

Do pracy ze skanerem bardzo przydatne jest oprogramowanie rozpoznające tekst — oprogramowanie typu OCR (*Optical Character Recognition*), stanowiące też standardowe wyposażenie wspomnianych tabletów.

Jest ono szczególnie potrzebne przy skanowaniu dokumentów tekstowych. Ponieważ po bezpośrednim przeczytaniu takiego dokumentu jest on zapamiętywany jako cyfrowy obraz, rozmiar pamięci potrzebny na jego przechowywanie jest bardzo duży, a ponadto nie ma bezpośredniej możliwości modyfikacji tekstu. Program typu OCR analizuje cyfrowy obraz, rozpoznaje występujące w nim znaki i na tej podstawie generuje plik tekstowy, który można przetwarzać tak jak każdy inny tekst.

Karta dźwiękowa

Powszechnym wyposażeniem komputera stają się obecnie urządzenia multimedialne: mikrofon, kamera i głośniki. Dołączenie ich do komputera wymaga po stronie komputera odpowiednich sterowników. Sterowniki mikrofonu i głośników są instalowane na karcie muzycznej, a sterowniki kamery — na karcie video.

Modem i karta sieciowa

Komputery osobiste są coraz częściej podłączane do Internetu — światowej sieci komputerowej. Praca w Internecie wymaga odpowiedniego wyposażenia w postaci karty sieciowej. Nowoczesne karty sieciowe są uniwersalne, pozwalają na dołączenie przez linię telefoniczną, telewizję kablową i przez połączenia sieciowe. Przez linie telefoniczne można prowadzić transmisję danych z szybkością 56 Kb/s, przez połączenia telewizji kablowej i sieci lokalnej — z szybkościami 10-100 Mb/s.

System nieprzerywalnego zasilania

Standardowo komputery są zasilane z sieci energetycznej. W przypadku komputerów domowych jest to zasilanie absolutnie wystarczające. Natomiast w przypadku komputerów obsługujących zakłady pracy o ruchu ciągłym, takie jak banki, wymagane jest zabezpieczenie na wypadek nawet chwilowych przerw w dostarczaniu energii. Najpowszechniej spotykanym zabezpieczeniem jest urządzenie zwane UPS (*Uninterruptible Power Supplier*). Jest to zasilacz bateryjny, który automatycznie podejmuje pracę w wypadku zaniku zasilania z sieci energetycznej. Istotną cechą zasilacza UPS jest, podobnie jak akumulatora, jego pojemność energetyczna, na podstawie której można wyliczyć okres zasilania danego komputera. Najprostsze urządzenia pozwalają tylko na krótki czas pracy (ok. 30 minut), w którym można ją zakończyć w sposób uporządkowany, to zna-

czy z zapamiętaniem stanu aktualnie wykonywanych zadań. Najbardziej rozbudowane urządzenia UPS nie tylko podtrzymują okresowe zasilanie, ale dodatkowo uruchamiają alternatywne źródła zasilania (agregaty prądotwórcze).

Komputer domowy nie wymaga urządzeń UPS, natomiast zaleca się podłączenie zasilania komputera przez deskę zasilającą z filtrem, który ogranicza od góry przepięcia (skoki napięcia powyżej nominalnego zakresu 240 V). W ten sposób zapobiega się ewentualnym awariom zasilacza komputera.

Elementy oprogramowania

Komputer przedstawia użytkową wartość tylko wówczas, gdy jest wyposażony w odpowiednie oprogramowanie. W uproszczeniu oprogramowanie komputera można podzielić na trzy hierarchicznie uporządkowane warstwy:

- oprogramowanie wbudowane,
- system operacyjny,
- oprogramowanie aplikacyjne.

Hierarchia pomiędzy warstwami opiera się na zależności usługodawca-usługobiorca. Każda warstwa położona niżej w hierarchii oferuje pewien zestaw usług warstwie położonej wyżej. Zatem warstwa niższa jest usługodawcą dla warstwy wyższej, a warstwa wyższa — usługobiorcą usług warstwy niższej.

Najniższą warstwą oprogramowania jest oprogramowanie wbudowane w sprzęt komputerowy. Oprogramowanie dla komputerów osobistych nazywa się systemem BIOS (*Basic Input Output System*). Jest on przechowywany w pamięci stałej na płycie głównej komputera. System uruchamia się automatycznie po włączeniu zasilania komputera. Celem działania tego oprogramowania jest rozpoznanie aktualnej konfiguracji sprzętowej i sprawdzenie gotowości poszczególnych elementów sprzętowych, a następnie przekazanie pracy oprogramowaniu, które znajduje się w warstwie wyższej.

Następną warstwę oprogramowania stanowi system operacyjny. Jest on zasadniczą częścią oprogramowania, które organizuje pracę komputera. System BIOS jest specyficzny dla danego typu płyty głównej i opracowywany przez producenta sprzętu. Wraz ze sprzętem system BIOS wyzna-

cza standardową platformę, na której można umieścić i umożliwić funkcjonowanie systemu operacyjnego.

System operacyjny jest pośrednikiem pomiędzy komputerem a jego użytkownikami. Jest on programem, albo precyzyjniej: kolekcją programów. Jego zadaniem jest realizacja poleceń kierowanych do komputera przez jego użytkowników. Do realizacji zleceń użytkowników system operacyjny wykorzystuje elementy składowe konfiguracji komputera: procesor, pamięci, urządzenia zewnętrzne, oraz zasoby informacyjne: pliki zawarte w pamięci komputera.

Podstawowe funkcje wykonywane przez system operacyjny obejmują następujące grupy czynności:

- obsługę komunikacji z użytkownikiem (klawiatura, mysz, monitor),
- obsługę innych urządzeń zewnętrznych (drukarka, modem),
- zarządzanie pamięcią operacyjną i dyskową (przydzielanie i zwalnianie miejsca w pamięci dla wykonywanych aplikacji, okresowe porządkowanie pamięci — odświeżanie),
- zarządzanie procesorem (przydział dostępu do procesora wykonywanym aplikacjom).

Najczęściej spotykamy dwie rodziny systemów operacyjnych — *Windows* i *Unix*. Każdy z tych systemów występuje w licznych, ciągle ewoluujących odmianach i wersjach. Systemy te funkcjonują we wszystkich komputerach osobistych zaliczanych do rodziny komputerów IBM PC (*International Business Machines Personal Computer*). Komputery tej rodziny oparte są na standardzie architektury opracowanej przez firmę IBM na początku lat osiemdziesiątych.

System operacyjny umożliwia dostęp do usług następnej warstwy oprogramowania, a także dostarcza pewnych własnych usług użytkownikom komputera. Do usług tych należą:

- otwieranie/zamykanie sesji użytkownika (logowanie/wylogowanie). Otwieranie sesji polega zwykle na podaniu identyfikatora użytkownika oraz hasła dostępu. Po otwarciu sesji użytkownik może korzystać z oprogramowania aplikacyjnego i operacyjnego. Nie wszyscy użytkownicy muszą mieć takie same prawa dostępu (administratorzy systemu mają prawo do korzystania ze wszystkich usług systemu operacyjnego, natomiast pozostali użytkownicy mogą korzystać tylko z części usług systemu operacyjnego, mają ograniczone możliwości dostępu do plików i wykonywanych na nich operacji);

- usługi katalogowe — obejmują przeglądanie katalogów (folderów, kartotek) i ich zawartości, tworzenie i kasowanie katalogów, poszukiwanie plików;
- wybór i uruchamianie aplikacji — umożliwia użytkownikowi wybór i uruchomienie dowolnego programu umieszczonego w pamięci komputera;
- usługi administracyjne — pozwalają na tworzenie grup użytkowników i przedzielanie im uprawnień;
- usługi konfiguracyjne — dotyczą instalowania i deinstalowania programów, ustalania parametrów urządzeń komputera — ekranu, klawiatury, ustawianie czasu i daty itd.

Następną warstwą jest warstwa oprogramowania aplikacyjnego obejmująca zbiór programów aplikacyjnych (krótko: aplikacji), które może wykorzystywać użytkownik komputera. Warstwa ta może funkcjonować tylko dzięki obecności systemu operacyjnego stanowiącego środowisko wykonawcze dla programów warstwy aplikacji. Jest to warstwa bardzo bogata, gdyż może zawierać dowolnie zestawione programy stosownie do potrzeb konkretnego użytkownika. W warstwie tej można wyróżnić przynajmniej trzy kategorie aplikacji:

- ogólnego przeznaczenia, np. pakiet *Microsoft Office* w środowisku systemu operacyjnego *Windows*,
- specjalistyczne, np. oprogramowanie do prowadzenia ksiąg handlowych, rozliczeń finansowo-księgowych, zintegrowanego zarządzania przedsiębiorstwem MRP (*Manufacturer Resource Planing*),
- narzędziowe, służące do wytwarzania innego oprogramowania, czyli tzw. środowiska projektowo-programistyczne.

Popularny pakiet *Microsoft Office* zawiera wiele aplikacji wystarczających do typowych potrzeb użytkownika. Są w nim zawarte m.in.:

- edytory: tekstowo-graficzny (*Word*), graficzny (*Photo Editor*),
- arkusz kalkulacyjny (*Excel*),
- program prezentacyjny (*Power Point*),
- system zarządzania bazami danych (*Access*),
- przeglądarka sieciowa (*Internet Explorer*),
- poczta elektroniczna (*Microsoft Outlook*).

Wymienione trzy warstwy oprogramowania — BIOS, system operacyjny, warstwa aplikacji — współpracują ze sobą. Warstwa niższa dostarcza usług, z których korzystać może warstwa wyższa. Współpraca ta jest też widoczna podczas uruchamiania komputera.

Po włączeniu zasilania natychmiast rozpoczyna pracę BIOS, który wykrywa elementy konfiguracji sprzętowej i przeprowadza ich diagnostykę, a następnie wprowadza do pamięci operacyjnej — z dyskietki lub dysku wewnętrznego — jądro systemu operacyjnego. Jądro po zainicjowaniu umożliwia użytkownikowi rozpoczęcie pracy, to znaczy otwarcie sesji (logowanie), a następnie wybór wykonywania dostępnych aplikacji — programów z warstwy aplikacji.

IV. SYSTEM PLIKÓW

Czym są pliki?

Dane i programy są przechowywane w pamięci komputera jako pliki. Plik jest jednostką zapisu i przechowywania danych w komputerze, jego zawartość ma postać ciągu bitów. Plik ma nazwę i atrybuty.

Mamy różne rodzaje plików. Podstawowy podział wyróżnia dwie kategorie plików: wykonywalne i niewykonywalne. Zawartością plików wykonywalnych są programy, niewykonywalnych — dane. Pliki niewykonywalne klasyfikuje się za względu na rodzaj przechowywanych danych. Stąd mówimy o plikach:

- tekstowych,
- graficznych,
- dźwiękowych,
- filmowych.

Często mamy pliki, które zawierają jednocześnie różne rodzaje danych — są to pliki wielomedialne. Kolekcje plików można wiązać ze sobą w całości — pliki hipertekstowe lub hipermedialne.

Pliki też rozróżnia się z punktu widzenia aplikacji, które je wykorzystują lub wytwarzają. Ze względu na różnorodność aplikacji trudno przedstawić systematyczną klasyfikację rodzajów plików.

Pliki różnych rodzajów stosują różne formaty zapisu, czyli kodowania danych. Każdy plik przechowywany w pamięci ma swój identyfikator — *nazwę*. Powszechnie przyjmuje się, że nazwa pliku jest napisem, który składa się z dwóch części: *nazwy głównej* i *rozszerzenia*, przedzielonych kropką. Nazwa pliku może jednak składać z wielu części. Nazwa główna może być ustalana dowolnie przez użytkownika, natomiast rozszerzenie służy określeniu rodzaju pliku. W obrębie każdej kategorii plików spotykamy różne rozszerzenia. Oto przykładowe nazwy plików:

```
mójProgram.exe  
mojeDane.txt
```

Nazwami głównymi są `mójProgram` oraz `mojeDane`, natomiast `exe` oraz `txt` są rozszerzeniami.

Rozszerzenie `exe`, a także inne, jak: `com`, `ddl`, oznaczają, że plik jest wykonywalny. Rozszerzenie `txt` oznacza, że plik jest niewykonywalny, a jego zawartością jest tekst.

Plikom przechowywanym w pamięci przypisuje się pewne atrybuty, takie jak:

- data i czas utworzenia (lub ostatniej modyfikacji) pliku,
- rozmiar pliku,
- określenie dopuszczalnych operacji wykonywanych na pliku, np. tylko czytaj, czytaj lub pisz do pliku.

Pliki przechowywane w pamięci komputera są grupowane w katalogi (foldery).

Pliki tekstowe i tekstowo-graficzne

Pliki tekstowe zawierają ciąg znaków z określonego repertuaru znaków. Każdy znak jest reprezentowany przez ciąg bitów. Repertuar znaków wraz ze sposobem kodowania binarnego nazywamy kodem znakowym. Wśród wielu kodów warto przypomnieć wcześniej omawiane kody ASCII oraz UNICODE. Zgodnie z polskimi normami technicznymi obowiązującym sposobem kodowania tekstów pisanych po polsku jest 8-bitowy kod ISO-Latin-2, nazywany również ISO-8859-2. Kod obejmuje wszystkie znaki alfabetu polskiego, a także innych języków słowiańskich bazujących na alfabecie łacińskim, oraz znaki alfabetu niemieckiego, węgierskiego i rumuńskiego.

Pliki tekstowe są zwykle zaznaczane przez rozszerzenie nazwy `txt`. Programy, które wytwarzają pliki tekstowe, nazywane są *edytorami*. Pozwalają one użytkownikowi na tworzenie, modyfikowanie i zapamiętywanie, a także drukowanie dokumentów tekstowych. Edytor tekstowy niezbędny jest m.in. do przygotowania tekstu programu.

Pierwotne pojęcie dokumentów tekstowych z czasem uległo rozszerzeniu. Obecnie dokumenty wytworzone za pomocą edytorów, takich jak *Word* z pakietu *Microsoft Office* (rozszerzenie `doc`), określane nominalnie jako tekstowe, oprócz samego tekstu zawierają np. tabele, wykresy, wzory matematyczne. Pociągnęło to za sobą konieczność opracowania nowych rozszerzonych formatów. Przykładami szeroko rozpowszechnionych formatów zawierających teksty wraz grafiką są: *Portable Document*

Format (pdf) oraz *PostScript* (ps). Z tego względu wiele edytorów posiada mechanizmy transformacji własnego specyficznego formatu na format pdf lub ps.

Pliki graficzne

Wyróżnia się dwa sposoby reprezentacji obrazów: rastrową (cyfrową) i wektorową.

Digitalizacja obrazu w grafice rastrowej polega na podziale obrazu na piksele. W takiej właśnie postaci obraz jest przedstawiany na ekranie monitora lub na wydruku z drukarki. Prostokąt, w którym zawiera się obraz, jest podzielony liniami poziomymi (rzędu 400-800) i pionowymi (rzędu 600-1200) na piksele. Pojedynczy piksel na całej swojej powierzchni ma taki sam kolor.

Digitalizacja, chociaż jest uproszczeniem obrazu, daje całkiem dobre efekty, gdyż oko ludzkie ma ograniczoną rozdzielczość i nie dostrzega, że oglądany obraz składa się z kolorowych punkcików, podobnie jak przy oglądaniu wystaw malarskich nie można z daleka dostrzec śladów pociągnięć pędzlem na olejnym obrazie.

Kolor piksela jest zapamiętywany jako liczba binarna. Zwykle do zapisu koloru używa się 1 lub 2 bajtów, co oznacza odpowiednio $2^8 = 256$ lub $2^{16} = 65536$ kolorów. Pojedynczy obraz o średniej jakości — 600 × 800 pikseli, przy rozróżnianiu 256 kolorów — wymaga do zapamiętania 3750 KB. Pamięć ta nie zależy przy tym od tego, co obraz przedstawia.

Do reprezentacji koloru piksela używa się różnych systemów. Najbardziej popularny jest system RGB (*red – green – blue*), gdzie każdy kolor jest przedstawiany w postaci mieszanki kolorów podstawowych — czerwonego, zielonego i niebieskiego.

System RGB jest powszechnie stosowany w grafice ekranowej, a także przy wydruku, choć w tym przypadku stosuje się także inne systemy, np. CMYK (*cyan – magenta – yellow*), który jako podstawowe przyjmuje kolory turkusowy, fioletowy i żółty.

Oprócz wymienionych stosowane są także inne systemy, m.in. ERGB (*Enhanced RGB*) oraz HSV lub HSB (*Hue, Saturation, Value/Brightness*), czyli odcień, nasycenie, jasność.

Ponieważ zapis obrazu w postaci rastrowej wymaga dużej pamięci, opracowano różne sposoby kompresji plików graficznych. Kompresja oznacza, że obraz oryginalny przed zapisem w pamięci komputera przetwarza

się tak, aby zredukować rozmiar pamięci potrzebnej do jego zapamiętania. Wyróżnia się dwie kategorie kompresji: bezstratną i stratną. Bezstratny sposób kompresji oznacza, że obraz skompresowany przy odtwarzaniu (wyświetlaniu lub wydruku) ma dokładnie taką samą postać jak obraz oryginalny. Natomiast kompresja stratna oznacza, że obraz odtworzony po skompresowaniu nie jest identyczny, ale zmiany w obrazie nie powinny być istotne, najlepiej gdyby nie były rozróżnialne przez człowieka. Oczywiście, nie wszystkie metody kompresji stratnej spełniają takie wymagania.

Stąd mamy różne formaty i rozszerzenia nazw tych plików, najczęściej spotykane to: *bmp* (*bitmap*) — zapis bez kompresji, *gif* (*Graphics Interchange Format*) — zapis z kompresją bezstratną, *jpeg* (*Joint Photo Expert Group*) — zapis z kompresją stratną.

Niektóre graficzne formaty rastrowe (np. *gif*) pozwalają na umieszczanie w pojedynczym pliku wielu „klatek”, które — oglądane kolejno — stwarzają wrażenie ruchu. W prostej animacji płynnego ruchu kolejne klatki są do siebie bardzo podobne, co można wykorzystać w odpowiedniej metodzie kompresji danych. Dlatego profesjonaliści korzystają ze specjalnych formatów zapisu obrazu ruchomego, takich jak *mpeg* i *avi*, które mają również charakter rastrowy.

Digitalizacja obrazu w grafice wektorowej polega na złożeniu całego obrazu z pewnego zestawu elementarnych obiektów, np. linii (prostych lub krzywych), wielokątów, kół, i ich rozmieszczeniu na pewnym tle. Każdy elementarny obiekt ma stałą reprezentację cyfrową. Do ich zapamiętania nie potrzeba tak dużej pamięci jak w przypadku reprezentacji rastrowej. Np. do zapamiętania odcinka wystarczą: współrzędne jego końców, grubość i kolor. Do zapamiętania obrazu, który zawiera tylko jeden odcinek, potrzebny jest dodatkowo kolor tła, razem mniej niż 100 bitów! Oczywiście, w przypadku bardziej złożonych obrazów będzie potrzebna większa pamięć.

Reprezentacji wektorowej używa się przy przechowywaniu obrazu w pamięci masowej lub podczas przesyłania w sieci komputerowej. Natomiast przy wyświetlaniu lub wydruku obraz w postaci wektorowej jest przekształcany w postać rastrową. Obraz przedstawiany na urządzeniu (monitor, drukarka, ploter) jest „kreślony” element po elemencie. Każdy element obrazu jest opisany za pomocą pewnej liczby cech (atrybutów), których wartości można zmieniać podczas edycji. Cechy — ich nazwy, właściwości i zasady edycji — zależą od środowiska, w którym powstaje

obraz. Poszczególne elementy obrazu mogą się wzajemnie przesłaniać lub przenikać. Przykładami formatów grafiki wektorowej są:

- *Computer Graphics Metafile* — do prezentacji obiektów dwuwymiarowych (rozszerzenie *cgm*),
- *AutoCAD* — do prezentacji obiektów trójwymiarowych (rozszerzenie *dgn*),
- *Scalable Vector Graphics* — do prezentacji obiektów dwuwymiarowych na stronach internetowych (rozszerzenie *svg*),
- *Portable Network Graphics*, zapis z kompresją (rozszerzenie *png*), stosowany m.in. do zapisu zdjęć w cyfrowych aparatach fotograficznych.

Dla zapisu plików czysto graficznych o wysokich wymagach dotyczących rozdzielczości, wierności oddania odcieni szarości lub kolorów przy powiększaniu lub zmniejszaniu obrazu opracowano wiele standardów. Standardy te obejmują definicję formatów, a także sposobów kompresji. Najczęściej spotykane obecnie formaty to *Joint Photographic Experts Group* oraz *Graphics Interchange Format* (rozszerzenia odpowiednio *jpeg* lub *jpg* oraz *gif*). Standard JPEG został zaprojektowany do kompresji i dekompresji obrazów kolorowych albo obrazów o zróżnicowanym stopniu szarości. Nie jest natomiast przeznaczony dla obrazów czarno-białych ani do obrazów ruchomych.

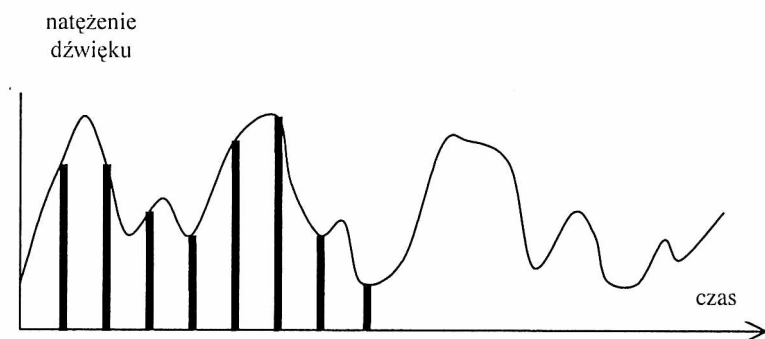
Odpowiednikami edytorów tekstowych są w omawianym przypadku edytory graficzne. Mamy tu do czynienia z szeroką gamą rozwiązań, od najprostszych stanowiących rozszerzenia edytorów tekstowych po edytory graficzne zintegrowane z różnymi systemami projektowania wspomagane komputerowo. Przykładem systemów projektowania jest rodzina tzw. systemów CAD (*Computer Added Design*). Specjalizacje tych systemów służą do projektowania różnych klas urządzeń, np. do projektowania architektonicznego, konstrukcji maszynowych, aranżacji i oświetlenia wnętrz.

Pliki audio (dźwiękowe)

Źródło dźwięku generuje falę, którą można przedstawić w postaci funkcji natężenia dźwięku. Zapamiętanie dźwięku oznacza zapamiętanie tej funkcji. Funkcję tę poddaje się digitalizacji polegającej na kwantyzacji czasu i dyskretyzacji natężenia. Kwantyzacja oznacza podział osi czasu na równe odcinki (kwanty), a dyskretyzacja — podział przedziału natężenia dźwięku na pewną liczbę odcinków (próbek). Każdemu z kwantów czasu

przypisuje się jedną z dyskretnych wartości natężenia, dającą się reprezentować liczbą binarną o ustalonej liczbie bitów. Typowym standardem jest kwantyzacja czasu z częstotliwością 44,1 kHz i podział przedziału natężenia na liczbę próbek określonych 16-bitową liczbą. Zawartością pliku dźwiękowego jest ciąg próbek dźwięku. Oznacza to, że do zapamiętania 1 sekundy dźwięku potrzeba ok. 1,4 Mb.

Rys. 4.1

Ilustracja kwantyzacji dźwięku

Podobnie jak w przypadku digitalizacji obrazów, digitalizacja dźwięków jest akceptowana dzięki ograniczeniom ludzkiego słuchu. Ograniczenia te odnoszą się do zakresu częstotliwości słyszalnych przez człowieka — od kilku Hz do kilkunastu KHz, oraz do wrażliwości na zmianę natężenia dźwięku. Przedstawiony standard w znacznym nadmiarze pokrywa wymagania ludzi o najlepszym nawet słuchu.

Digitalizację dźwięku stosuje się także w innych dziedzinach, np. w telefonii cyfrowej, przy czym używa się tam standardów o niższej jakości.

Podstawowym problemem przy zapisie dźwięku jest ogromne zapotrzebowanie na pamięć. Ma to szczególne znaczenie przy przesyłaniu dźwięku w Internecie, gdyż pociąga za sobą duże wymagania dotyczące przepustowości linii transmisyjnych. Podobnie jak w przypadku obrazów, opracowano mechanizmy kompresji plików dźwiękowych. Powszechnie stosowane metody kompresji stratnej polegają na takiej eliminacji części informacji zawartej w pliku dźwiękowym, by odbiorca nie mógł odróżnić dźwięku powstałego z odtwarzania pliku skompresowanego od dźwięku powstałego z odtwarzania pliku nieskompresowanego.

Stosowany obecnie standard kompresji MP3, opracowany przez *Moving Picture Expert Group*, pozwala na 12-krotne zredukowanie rozmiaru plików. Pliki zapisane zgodnie z tym standardem mają rozszerzenie mp3.

Znane są też inne standardy, np. system *waveform* (rozszerzenie wav) stosowany przez Microsoft; określa on współpracę przez interfejs MIDI (*Musical Instrument Digital Interface*) i został opracowany dla potrzeb zespołów muzycznych.

Tworzenie plików dźwiękowych może polegać na rejestrowaniu naturalnych źródeł dźwięku (nagrywanie rozmowy, występów muzycznych), ale także na sztucznym ich wytwarzaniu za pomocą specjalistycznego oprogramowania stanowiącego wyposażenie studiów muzycznych. Z punktu widzenia niezaawansowanego użytkownika komputera przydatny jest tylko program pozwalający na odtworzenie plików muzycznych. Programów takich jest wiele, np. *Quick Time Player* wchodzący w skład systemu *Microsoft Windows*.

Pliki video (filmowe)

Digitalizacja obrazu ruchomego polega na przedstawieniu go jako ciągu statycznych obrazów graficznych. Takie podejście jest stosowane od zainicjowania kinematografii i telewizji. Współczesny standard kinowy polega na wyświetlaniu 24 klatek (obrazów) na sekundę, a telewizji analogowej — 25 klatek na sekundę. Dzięki bezwładności ludzkiego oka ciąg wyświetlanych obrazów daje wrażenie ruchu ciągłego. Standardy te nie są najwyższej klasy, o czym można się przekonać po zauważalnym drganiu obrazu filmowego lub telewizyjnego oglądanego z boku kątem oka. W przypadku ruchomego obrazu wyświetlanego na ekranie monitora unika się tego efektu dzięki zwiększeniu liczby wyświetlanych obrazów (częstotliwości odtwarzania) do ponad 70 na sekundę.

Ze względu na ogromne znaczenie obrazu ruchomego w zastosowaniach informatycznych, w tym internetowych, oraz ich powiązań z kinematografią i telewizją cyfrową, od początku lat dziewięćdziesiątych w ramach ISO są prowadzone prace nad standardami reprezentacji i kompresji obrazu ruchomego. Prace te dodatkowo uwzględniają powiązania obrazu z dźwiękiem i animacją komputerową. Efektem prac jest seria standardów o nazwie MPEG (od *Moving Picture Experts Group* — grupy roboczej działającej w ramach ISO), m.in.: MPEG-1, MPEG-2, MPEG-4; kolejne są w przygotowaniu. Standard MPG-2 służy przesyłaniu obrazów

telewizyjnych i video na DVD, a MPG-4 — telefonii obrazowej i w sieci komputerowej. Pliki filmowe lub video, zapisane w standardzie MPEG, rozpoznaje się po rozszerzeniu *mpg* lub *mp*.

Podobnie jak w przypadku plików muzycznych, pliki video można tworzyć przez nagrywanie kamerą cyfrową naturalnych obrazów i scen, ale także na sztucznym ich wytwarzaniu za pomocą specjalistycznego oprogramowania stanowiącego wyposażenie studiów telewizyjnych. Niezaawansowany użytkownik komputera potrzebuje tylko programu odtwarzającego pliki filmowe, np. *Windows Media Player*, wchodzący w skład systemu *Microsoft Windows*.

Pliki hipertekstowe i hipermedialne

Odrębną grupą są pliki hipertekstowe, powszechnie wykorzystywane w Internecie. Wszelkie serwisy informacyjne oparte są współdziałaniu systemów komputerowych gromadzących dokumenty hipertekstowe.

Dokumenty hipertekstowe są uogólnieniem pierwotnym dokumentów tekstowych. Uogólnienie polega na tym, że dany dokument tekstowy dzieli się na powiązane ze sobą części składowe. Poszczególne części mogą zawierać w swojej treści odsyłacze (*łączniki*) do innych części. Podczas przeglądania dokumentu hipertekstowego odsyłacz można wykorzystać do przeskoku z danego fragmentu dokumentu do innego, np. poprzez wskazanie wyróżnionego słowa lub zwrotu można wyświetlić bardziej szczegółowe informacje na dany temat.

Odsyłacze w dokumencie mogą prowadzić nie tylko do innych fragmentów danego dokumentu, ale także do innych dokumentów, nawet ulokowanych w innych komputerach.

Najpowszechniej używanym współcześnie formatem plików hipertekstowych jest format HTML (*Hypertext Markup Language*). Pliki o tym formacie mają rozszerzenie *html* lub *htm*. Format HTML jest jednak obecnie zastępowany nowym formatem XML (*Extensible Markup Language*).

Uogólnieniem plików hipertekstowych są pliki hipermedialne, które dopuszczają, że oprócz tekstu dokument może zawierać również obrazy, dźwięk i film. W przypadku plików hipermedialnych wskazanie odsyłacza na ekranie monitora może prowadzić nie tylko do wyświetlenia innych tekstów, ale i rysunków, odtworzenia dźwięków lub klipów filmowych (krótkich filmów).

Pliki skompresowane

Ze względu na trudności powodowane przez duże rozmiary plików często stosuje się kompresję plików. Kompresja jest redukcją rozmiaru pliku przez zmianę formatu przechowywanych w nim danych. Najczęściej stosowane algorytmy eliminują powtarzające się ciągi bitów, zapisując je w krótszej postaci. Odczytanie skompresowanego pliku możliwe jest po przeprowadzeniu procesu odwrotnego, zwanego dekompresją.

Przesyłanie przez Internet skompresowanych zbiorów pozwala zaoszczędzić czas i koszty transmisji. Dość często stosowaną metodą jest kompresja w locie, to znaczy, że przed transmisją plik jest kompresowany, wysyłany w postaci skompresowanej, a po otrzymaniu przywracany do formatu oryginalnego.

Metody kompresji bezstratnej pozwalają spakować, a następnie odtworzyć plik bez zagubienia informacji. Wykorzystuje się je do kompresji dokumentów tekstowych, baz danych, programów — wszędzie tam, gdzie zmiana pojedynczego bitu może zaważyć na znaczeniu całego zbioru (np. program może w ogóle nie dać się uruchomić). Przykładami są formaty: zip, gzip, arj, rar, tar, za którymi kryją się odpowiednie programy kompresujące.

W przypadku plików graficznych, dźwiękowych i filmowych zwykle nie ma potrzeby używania programów kompresujących, gdyż wytworzone pliki są przedstawione w skompresowanych formatach, np. JPEG, MPEG, MP3. Jak wspominaliśmy wcześniej, do tego rodzajów plików stosuje się metody kompresji stratnej, co pozwala na większy stopień upakowania danych (nawet do dziesiątej części oryginału). Spowodowana tym utrata pewnej informacji jest możliwa do zaakceptowania tylko dzięki ograniczoności zmysłów człowieka.

Pliki w systemie Windows

Poza omówionymi wyżej plikami spotyka się pliki o innych formatach. Są one zwykle związane z danym systemem operacyjnym lub ze specyficznymi aplikacjami.

Przykładem aplikacji związanych z systemem operacyjnym *Microsoft Windows*, w grupie określonej jako akcesoria, są m.in.:

- *Notatnik* — prosty edytor tekstowy (pliki o rozszerzeniu `.txt`),

- *Paint* — prosty edytor grafiki rastrowej (pliki o rozszerzeniu *bmp*),
- edytor *WordPad* — uproszczona wersja edytora *Word* (pliki o rozszerzeniu *rtf*).

Przykładem specyficznej aplikacji jest pakiet *Microsoft Office*, stanowiący zestaw kilku aplikacji. Głównymi aplikacjami pakietu są: *Word*, *Power Point*, *Excel*, *Access*, które tworzą pliki o specyficznych rozszerzeniach.

Edytor *Word* generuje pliki o rozszerzeniu *doc*. Zawartością tych plików jest nie tylko tekst, ale również tabele, wzory matematyczne, rysunki, obrazy. Wyrazem możliwości edytora jest to, że za jego pomocą można przygotować artykuły, raporty, a nawet książki.

Innego rodzaju edytorem jest program *Power Point*, który tworzy pliki o rozszerzeniu *ppt*. Program ten spełnia dwie funkcje: edytora plików oraz organizatora pokazu slajdów. W odróżnieniu od edytora *Word*, który wytwarza pliki przeznaczone ostatecznie do wydruku, *Power Point* przygotowuje pliki głównie do pokazu slajdów. Zawartością plików wytwarzanych przez ten program jest nie tylko tekst i grafika, ale również elementy prostej animacji obrazu.

Arkusz kalkulacyjny *Excel* to program do prowadzenia obliczeń na podstawie danych zestawionych w tabele, używany zwłaszcza w rachunkowości; generuje on pliki o rozszerzeniu *xls*.

System zarządzania bazą danych *Access* tworzy pliki o rozszerzeniu *dbf*. Baza danych jest pewnym rodzajem zbioru danych związanych z fragmentem interesującej nas rzeczywistości. Posiada mechanizmy pozwalające na modyfikowanie zgromadzonego zbioru danych oraz mechanizmy wyszukiwania w tym zbiorze danych o interesujących nas właściwościach.

Różne aplikacje podczas pracy tworzą tymczasowe pliki robocze, które zwykle przed zakończeniem pracy są kasowane. Najczęściej pliki takie można rozpoznać po rozszerzeniu *tmp*. Zdarza się, że z różnych powodów — niedoskonałości aplikacji lub zakłóceń podczas jej pracy — pliki utworzone jako tymczasowe nie są usuwane, a przy tym po kolejnym uruchomieniu aplikacji nie są przez nią wykorzystywane. Może to stać się przyczyną zaśmiecania pamięci. Dlatego użytkownik komputera powinien okresowo sprawdzać i usuwać niepotrzebne pliki tymczasowe.

Kartoteki plików

Liczba plików przechowywanych w komputerze osobistym jest zwykle bardzo duża, wynosi co najmniej kilkanaście tysięcy. Panowanie nad tak licznym zbiorem, to znaczy znajdowanie i sięganie do zawartości potrzebnych plików, wymaga odpowiedniego ich uporządkowania.

Do tego służą kartoteki, inaczej: foldery lub katalogi, wyznaczające podział zbioru wszystkich plików na pewne podzbiory. Podział na kartoteki może być ustalony dowolnie przez użytkownika komputera. Zaleca się jednak, by niektóre kartoteki nazywać i używać zgodnie z podziałem ustalonym przez system operacyjny komputera. System operacyjny, który kupujemy zwykle z komputerem, wprowadza własne kartoteki, przeznaczone do przechowywania oprogramowania podstawowego i przygotowuje pewne kartoteki dla plików użytkownika komputera.

Zawartością kartoteki jest zwykle pewien podzbiór plików, a dodatkowo w obrębie danej kartoteki można definiować jej kartoteki wewnętrzne — podkartoteki. Podkartoteki, podobnie jak kartoteki, w których są zawarte, mogą mieć swoje podkartoteki, a podkartoteki podkartotek jeszcze swoje podkartoteki itd. Kartoteka, która nie jest zagnieżdżona w innej kartotece, nazywa się kartoteką główną albo kartoteką-korzeniem, a kartoteka, w której nie są zagnieżdżone inne kartoteki, nazywa się kartoteką-liściem. Użyte terminy wynikają stąd, że zestaw wszystkich kartotek stanowi pewną hierarchiczną strukturę, graficznie reprezentowaną przez graf-drzewo. Zestaw wszystkich kartotek wraz z ich zawartością w pamięci komputera nazywamy *systemem plików*.

System plików komputera jest zwykle zdekomponowany na systemy plików związane z pojedynczymi pamięciami trwałymi — wewnętrznymi dyskami twardymi oraz z pamięciami zewnętrznymi — dyskietskami, płytami CD, pamięciami *flash* itp.

Kartoteki można przeglądać dzięki specjalnym programom, zawartym w każdym systemie operacyjnym, np. w systemach *Microsoft Windows* jest to program zarządzania kartotekami i plikami o nazwie *Explorer*. W kartotece o nazwie *Mój komputer* pojawiają się kartoteki zwykle oznaczane początkowymi literami alfabetu łacińskiego A:, C:, D: itd., które odnoszą się do różnych pamięci trwałych. Nazwa A: odnosi się do pamięci dyskietkowych, C: i D: — do wewnętrznych dysków twardych, a nazwy z kolejnymi literami — do płyt CD, DVD, pamięci *flash* itd.

Pliki pamiętane w danym systemie plików są identyfikowane przez nazwy i kartoteki, w których się znajdują. Pliki w obrębie danej kartoteki muszą mieć unikalne nazwy. Te, które mają jednakowe nazwy, ale znajdują się w różnych kartotekach, są traktowane jako różne pliki. Zatem sama nazwa pliku nie stanowi jego jednoznacznej identyfikacji. Plik jest jednoznacznie identyfikowany przez *ścieżkę dostępu*, czyli nazwę pliku poprzedzoną ciągiem nazw kolejnych kartotek, w których jest zagnieżdżony. Np. dla pliku o nazwie `arj.exe`, zapamiętanego na dyskietce, w kartotece archiwizator jego ścieżka dostępu ma postać:

`A:/archiwizator/arj.exe`

Inny przykład ścieżki dostępu:

`C:/MojePliki/archiwizator/arj.exe`

oznacza, że plik o tej samej nazwie `arj.exe` jest przechowywany na dysku twardym `C:`, w podkartotece `archiwizator` kartoteki `MojePliki`.

Symbole ukośnika / są separatorami przedzielającymi kolejne nazwy w ścieżce dostępu.

Na system plików składają się nie tylko pliki i kartoteki, ale także operacje, jakie można na nich wykonywać, np.:

- tworzenie, kasowanie i zmiana nazw kartotek,
- kasowanie, przenoszenie, kopiowanie i zmiana nazw plików.

Ponieważ często z tego samego komputera korzysta kilku użytkowników, można im nadać różne prawa dostępu i operowania na plikach zapisanych w komputerze. Prawa użytkownika określa się przez wskazanie kartotek, do których ma on dostęp, i operacji, jakie może wykonywać na udostępnionych mu kartotekach.

Różne systemy operacyjne wprowadzają szczegółowe mechanizmy nadawania tych praw, ale wspólnym ich elementem jest podział użytkowników na dwie kategorie: administratorów i zwykłych użytkowników. Administratorzy mają nieograniczone prawa dostępu do wszystkich plików, a ponadto dysponują mechanizmami nadawania praw dostępu dla zwykłych użytkowników.

Z systemem plików jest związany mechanizm ich wyszukiwania. Jest to program, który na podstawie pewnych informacji o poszukiwanym pliku (lub kartotece) przeszukuje pamięć komputera i wskazuje wszystkie te pliki (lub kartoteki), które posiadają wskazane własności. Najczęściej informacjami, które dotyczą poszukiwanych plików, mogą być ich nazwy bądź fragmenty nazw. Dodatkowo mogą to być niektóre atrybuty

plików, np. czas, w którym plik został utworzony. W przypadku poszukiwania plików tekstowych można wskazać fragment tekstu, który poszukiwany plik powinien zawierać.

V. INTERFEJS KOMUNIKACYJNY

Wprowadzenie

Opanowanie złożoności współczesnych systemów informatycznych jest możliwe dzięki stosowaniu zasady dekompozycji. Dekompozycja funkcjonalna polega na podziale funkcji wykonywanych przez cały system na warstwy funkcjonalne. Warstwy są uporządkowane hierarchicznie. Po między dwiema sąsiadującymi warstwami zachodzi relacja usługodawca-usługobiorca. Warstwa niższa — usługodawca oferuje swoje usługi warstwie wyższej — usługobiorcy.

W uproszczeniu w każdym systemie komputerowym można wyróżnić przynajmniej trzy warstwy:

- warstwa sprzętowa, na którą składają się urządzenia cyfrowe wraz z wbudowanym oprogramowaniem (np. BIOS),
- warstwa systemu operacyjnego,
- warstwa aplikacyjna, z której bezpośrednio korzysta użytkownik.

Użytkownika komputera można także potraktować jako element czwartej warstwy, która korzysta z usług komputera, a która być może świadczy usługi innym ludziom. Użytkownik różni się tym od pozostałych warstw, że może korzystać z usług dwóch warstw — systemu operacyjnego i aplikacji.

Do realizacji usług konieczna jest komunikacja między warstwami, a także między użytkownikiem komputera a jego funkcjonalnymi warstwami. Usługobiorca składa usługodawcy zamówienie usługi, a usługodawca składa sprawozdanie z jej realizacji.

W przypadku komunikacji z systemem komputerowym użytkownik kieruje pewne polecenie, a system je wykonuje, po czym przekazuje komunikat z wykonania polecenia. Polecenie określa operację, którą należy wykonać oraz ewentualnie określa dane (parametry) do tej operacji. Odpowiedzią systemu jest sprawozdanie z wykonania zleconej operacji. Może ono zawierać wynik operacji lub informację o tym, że operacja została

wykonana poprawnie bądź nie, z ewentualnym podaniem dalszych szczegółów. Często przyjmuje się zasadę, że brak odpowiedzi systemu oznacza pomyślną realizację powierzonego zlecenia.

Pomiędzy warstwami istnieją *interfejsy komunikacyjne*. Oznacza to, że warstwy są ze sobą powiązane — istnieje pomiędzy nimi kanał komunikacji, którym są przesyłane komunikaty. Komunikaty są wyrażane w języku służącym do formułowania zleceń usług i wyników ich realizacji. Język ten obejmuje pewien zestaw symboli, w postaci tekstowej lub graficznej, którym jest przypisane znaczenie. Komunikaty wymieniane pomiędzy warstwami powinny być elementami tego języka, oraz powinny być wymieniane w odpowiedniej (sensownej) kolejności. Komunikacja może mieć rozbudowaną postać dialogu, który składa się z ciągu poleceń i odpowiedzi, przy czym w kolejno formułowanych poleceniach i odpowiedziach może być jawne lub niejawne odniesienie do elementów przekazywanych we wcześniejszej części dialogu.

Interfejs pomiędzy użytkownikiem a komputerem, dokładniej pomiędzy użytkownikiem i systemem operacyjnym oraz użytkownikiem i warstwą aplikacji, nazwa się *interfejsem użytkowym*, natomiast interfejsy pomiędzy pozostałymi warstwami nazywa się *interfejsami programowymi*.

Pojęcie *interfejsu* jest szersze od przedstawionego wyżej, gdyż może odnosić się również do sprzężenia pomiędzy urządzeniami technicznymi — mówimy wówczas o *interfejsie sprzętowym*. Przykładem takich interfejsów są połączenia monitora, klawiatury, drukarki i innych urządzeń zewnętrznych z jednostką centralną komputera. Interfejs sprzętowy odnosi się przede wszystkim do ustalenia własności mechanicznych (np. kształt, rozmiary elementów łączówki) i własności elektrycznych (np. poziomy napięcia, częstotliwość sygnału) łączących się styków. Ustalenia te są niezbędne do zapewnienia wymiany sygnałów elektrycznych, natomiast interpretacja tych sygnałów i określenie kolejności ich wymiany należy do tzw. logicznych własności interfejsu.

Komunikaty wymieniane pomiędzy użytkownikiem a systemem na interfejsie użytkowym mają pewną formę i niosą pewną treść. Ze względu na formę można wyróżnić dwa zasadnicze rodzaje interfejsów:

- interfejs znakowy,
- interfejs graficzny.

Obecnie dominuje interfejs graficzny, który wyparł historycznie młodszy interfejs znakowy. Interfejs graficzny utożsamia się z *interfejsem okienkowym*. Wynika to z postaci obrazu na ekranie monitora przy komunika-

cji za pomocą interfejsu graficznego — komunikacja z komputerem i sterowanie programami odbywa się za pomocą okien, ikon, przycisków, suwaków i rozmaitych menu. Interfejs graficzny jest obsługiwany za pomocą myszy i sprzężonego z nią kursora, możliwe jest też używanie klawiatury.

Interfejs może dopuszczać różne formy komunikacji, może być np. głosowy i obrazowy (video), bądź też je łączyć — mówimy wówczas o *interfejsie multimedialnym*.

Interfejs głosowy (audio) oznacza, że wymieniane komunikaty mają postać dźwięków. Jest on stosowany od wielu lat np. w systemach militarnych. Poza ułatwianiem przekazywania poleceń taka forma, dzięki analizie i rozpoznaniu głosu, pozwala na identyfikację użytkownika. Interfejs głosowy pozwala na formułowanie poleceń w języku naturalnym. Obecnie upowszechnia się on w obsłudze telefonów komórkowych, np. przy wybieraniu numeru telefonu.

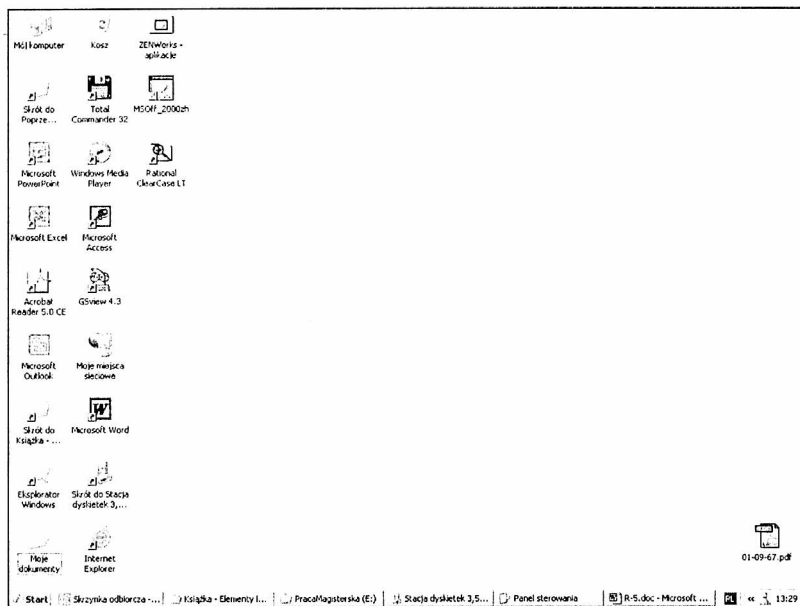
W przypadku komputerów osobistych interfejs obrazowy, czyli komunikacja z komputerem przez pokazywanie obrazów lub gestów, nie jest praktycznie używany. Są jednak systemy, w których występują pewne elementy takiego interfejsu, np. systemy dozoru wejścia do niektórych instytucji lub pomieszczeń; jedno z rozwiązań polega na identyfikacji osoby na podstawie odczytu tęczówki oka kamerą cyfrową.

Interfejs okienkowy

Interfejs okienkowy jest współcześnie standardem spotykanym praktycznie we wszystkich systemach. Podstawą do wydawania poleceń przez użytkownika są elementy graficzne umieszczone na ekranie monitora, które stanowią *pulpit roboczy*. Są to:

- ikony aplikacji,
- okienka aplikacji,
- paski narzędziowe,
- belki (linijki) menu, rozwijalne menu, podmenu,
- przyciski,
- okienka dialogowe zawierające różne rodzaje pól: pola sterujące (zatwierdzanie wyborów, rezygnacja), pola tekstowe, pola wyborów (wybieranie pojedynczych opcji, rozwijalne listy, wybieranie podzbiorów).

Rys. 5.1
Przykład wyglądu pulpitu roboczego

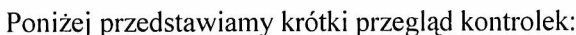


U dołu pulpitu znajduje się ciąg przycisków, z których pierwszy z lewej stanowi menu do usług systemu operacyjnego, a pozostałe są ikonami reprezentującymi otwarte aplikacje. Elementy graficzne powyżej są ikonami wybranych aplikacji, które są dostępne użytkownikowi.

Użytkownik wskazuje wybrane elementy, posługując się myszką lub klawiaturą i po akceptacji wyboru uruchamia reprezentowane przez nie akcje. Efektem podjętych akcji może być pokazanie się nowych okienek, które przedstawiają wyniki obliczeń lub stanowią dalszą część dialogu z użytkownikiem.

Okno aplikacji może posiadać w swoim wnętrzu różne elementy, np. własne paski narzędziowe, menu, przyciski, okienka dialogowe. Elementy te nazywa się kontrolkami, są nimi wszystkie elementy graficzne za wyjątkiem samego okna aplikacji.

Rys. 5.2



Przycisk

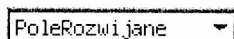
Przycisk (*button*) jest kontrolką zwykle o prostokątnym kształcie i szarym kolorze z czarnym napisem. Jest to najczęściej wykorzystywana kontrolka do inicjacji działań.

Polibory

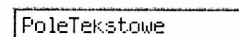
Przełącznik dwustanowy (*checkbox*) jest kontrolką, która może być zaznaczona lub odznaczona.

Polibor

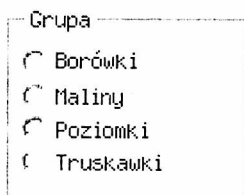
Przełącznik wielostanowy (*radio button*) pozwala na wybór jednej opcji spośród wielu.



Rozwijane listy (*combo*) dają możliwość wybrania jednej lub wielu wartości z listy dostępnych możliwych wartości.



Okienko tekstowe (*text*) służy przesyłaniu informacji od i do użytkownika.



Kontrolka grupowa (*group*) jest używana najczęściej do wizualnego pogrupowania powiązanych ze sobą kontroltek.



Menu jest kontrolką zwykle o kształcie listwy (belki) z listą nazw funkcji — pozycji menu. Jedno okno aplikacji może zawierać tylko menu. Za pojedynczą pozycją menu może kryć się rozwijana lista, której elementy określa się mianem podmenu. Podmenu może być dalej rozwijane w kolejne podmenu.



Kontrolka przełączająca pomiędzy oknami aplikacji (*TabOrder*) służy do wskazywania i przechodzenia do jednego z okien otwartych aplikacji.

W danym momencie spośród kontroltek widocznych na ekranie użytkownik może wybrać do wykonania tylko te, które są aktywne. Kontrolki aktywne odróżniają się od nieaktywnych tym, że są zwykle jaśniejsze.

Praca z komputerem

Otwieranie i zamykanie sesji roboczej

Pracę z komputerem, po włączeniu zasilania, rozpoczyna się od logowania, które polega na podaniu konta użytkownika i hasła. Efektem logowania jest uzyskanie dostępu do zasobów komputera, czyli programów i danych znajdujących się w pamięci komputera.

Konto określa zestaw zasobów dostępnych użytkownikowi i zakres operacji, które może na nich wykonywać. Zwykle stosuje się zasadę, że każdy użytkownik ma oddzielne konto, w tym przypadku konto jest nazwą użytkownika — nazwiskiem lub pseudonimem.

Hasło natomiast jest potwierdzeniem uprawnień osoby rozpoczynającej pracę do korzystania z konta. Ma ono postać ciągu znaków ustalonego przez użytkownika.

Zanim zwykły użytkownik się zaloguje, musi ustalić z administratorem komputera swoje konto i początkowe hasło. Kupując komputer, a ściślej system operacyjny, otrzymuje on konto administratora i początkowe hasło. Zarówno właściciel komputera, jako jego użytkownik i administrator, jak i zwykły użytkownik, po pierwszym logowaniu może zmienić początkowe hasło tak, by mieć gwarancję, że nikt nieuprawniony nie będzie miał dostępu do zasobów komputera. Dodatkowo administrator może zarejestrować innych użytkowników, przydzielając im konta z odpowiednimi uprawnieniami i hasłami początkowymi.

Po zalogowaniu użytkownik ma otwartą sesję, podczas której może prowadzić pracę z komputerem, kierując do niego odpowiednie polecenia. Po zakończeniu pracy następuje wylogowanie, czyli zakończenie sesji. Wylogowanie jest prostsze od zalogowania, gdyż polega tylko na skierowaniu do komputera odpowiedniego polecenia: *wyloguj* lub *wyloguj i wyłącz komputer* oraz na ewentualnym jego potwierdzeniu.

Logowanie, sesja robocza i wylogowanie przebiegają pod nadzorem systemu operacyjnego komputera. W trakcie roboczej sesji system operacyjny oferuje użytkownikowi różne usługi, do których należą:

- usługi kartotekowe,
- usługi administracyjne,
- wybór aplikacji do współdziałania.

Usługi kartotekowe oznaczają dostęp do systemu plików, czyli kartotek i zawartych w nich plików, oraz do wykonywania na nich różnych

operacji, np. kopiowania, kasowania, przenoszenia pomiędzy kartotekami, zmiany nazwy, otwierania.

Usługi administracyjne obejmują szeroki zakres funkcji. Są one dostępne przede wszystkim administratorom komputera, zwykli użytkownicy mogą z nich korzystać w bardzo ograniczonym zakresie. Do wyłącznych uprawnień administratorów, poza wspomnianym rejestrowaniem i wyrejestrowywaniem użytkowników, należą usługi konfiguracyjne: instalowanie i odinstalowywanie oprogramowania użytkowego oraz instalowanie i odinstalowywanie sprzętu. Przykładem usług konfiguracyjnych, dostępnych oddzielnie wszystkim użytkownikom, jest ustalanie parametrów ekranu (tła, kolorystyki elementów interfejsu graficznego, rozdzielczości i wygaszaczy ekranu), klawiatury (wybór języka) oraz myszy (konfigurowanie przycisków, ustalanie postaci kursora).

System operacyjny umożliwia wybór aplikacji, czyli programu. Dokonuje się tego przez wskazanie pliku, który zawiera potrzebny program. Po wybraniu jednej użytkownik może wybierać do wykonania następne aplikacje. Rozpoczęcie wykonywania aplikacji określa się też jako *otwarcie* aplikacji, a jej zakończenie — które może nastąpić automatycznie lub na polecenie użytkownika — jako *zamknięcie* aplikacji.

System operacyjny umożliwia też wykonywanie aplikacji — zarządza procesorem, przydzielając go wykonywanym aplikacjom, gospodaruje pamięcią i zarządza urządzeniami zewnętrznymi, przez które aplikacje komunikują się z użytkownikiem.

W komputerze może być wiele jednocześnie otwartych aplikacji. Aplikacje te mogą komunikować się z użytkownikiem przez interfejs użytkowy, zależny od rodzaju aplikacji. Jest to najczęściej interfejs okienkowy, ale jeszcze użytkuje się aplikacje, które powstały przed kilku laty i używają interfejsu znakowego.

Przykładowymi aplikacjami są programy pakietu *Microsoft Office*. Użytkownik może zatem mieć jednocześnie otwarte różne aplikacje tego pakietu, np. *Word* i *Excel*. Ponieważ wszystkie aplikacje pakietu mają interfejs okienkowy, oznacza to, że widocznym efektem tego będzie obecność na ekranie monitora dwóch okien odpowiadających dwóm otwartym aplikacjom.

Sesja robocza

Głównym celem pracy z komputerem jest wykonywanie wymaganych obliczeń, za pomocą wskazanych przez użytkownika programów aplikacyjnych. Aplikacje są reprezentowane przez ikony na pulpicie roboczym

lub przez swoje nazwy umieszczone w katalogach plików. Wybraną aplikację wskazuje się przez podświetlenie ikony lub nazwy, a następnie przez naciśnięcie klawisza *Enter*, przy posługiwaniu się klawiaturą, lub przez podwójne kliknięcie lewego przycisku myszy.

Otwarcie aplikacji, np. edytora *Word*, spowoduje pojawienie się nowej planszy na ekranie monitora — planszy roboczej edytora *Word*. Będzie na niej umieszczony zestaw elementów interfejsu graficznego służącego do pracy z edytorem. Korzystając z tego interfejsu, można do edycji wybrać jeden z istniejących plików lub utworzyć nowy plik.

Możliwy jest również inny sposób uruchamiania (otwierania²) aplikacji. Polega on nie na bezpośrednim jak wyżej, ale pośrednim wskazaniu aplikacji. Tak jest wówczas, gdy wskazuje się plik, który może być interpretowany przez pewną aplikację. Np. plik o rozszerzeniu nazwy *doc* w systemie *Windows* jest plikiem, który został utworzony przez edytor *Word*. Wskazanie tego pliku i naciśnięcie klawisza *Enter* lub podwójne kliknięcie myszą spowoduje podwójny efekt: po pierwsze zostanie otwarta aplikacja *Word*, a po drugie — zostanie otwarty wskazany plik, a jego zawartość będzie wyświetlona na ekranie monitora. Wyświetlony plik jest już otwarty do edycji.

Współczesne komputery pozwalają użytkownikowi na uruchamianie i jednoczesną pracę wielu aplikacji. System operacyjny organizuje pracę komputera w taki sposób, że użytkownik odnosi wrażenie jednoczesnego postępu obliczeń wszystkich aplikacji. Efekt ten uzyskuje się przez cykliczny podział czasu pracy procesora dla wybranych aplikacji.

W trakcie realizacji aplikacja może korzystać z plików umieszczonych w komputerze, może wytwarzać nowe pliki i umieszczać je w pamięci komputera, może także — na okres swojej realizacji — tworzyć tymczasowe pliki robocze. Dobrze zaprojektowana aplikacja przed normalnym zakończeniem usuwa swoje pliki robocze. Czasem zdarza się, że aplikacja jest kończona w sposób awaryjny, np. wskutek nieprzemyślanej akcji użytkownika, wówczas tymczasowe pliki robocze nie zostają usunięte, obciążając tym samym pamięć komputera. Usunięcia ich powinien dokonać użytkownik.

W systemie *Windows* pliki tymczasowe można rozpoznać po nazwie głównej i po rozszerzeniu. Nazwa główna zaczyna się symbolem ~ (tyl-

² Stosowane terminy: otwieranie (uruchamianie) i zamykanie (zatrzymywanie) aplikacji wiąże się z otwieraniem i zamykaniem okienek dialogowych skojarzonych z aplikacją. Stąd wynika powszechność ich użycia.

da), zaś rozszerzeniem jest tmp, np. ~DF21A4.tmp. Nazwy te wskazują tylko, że plik jest tymczasowy, dodatkową informacją potrzebną do stwierdzenia, czy plik można skasować, jest czas jego utworzenia. Może się zdarzyć, że przeglądając zawartość katalogów, zobaczymy nazwy roboczych plików aplikacji, które w danej chwili są wykonywane. System operacyjny zapobiega w takich przypadkach próbom ich skasowania przez komunikat, że plik jest aktualnie używany.

Ponieważ nieprzemyślane skasowanie pliku może przynieść czasem niepowetowaną utratę danych, operacja powinna być podejmowana świadomie. Polecenie kasowania nie jest wykonywane natychmiast, lecz wymaga potwierdzenia przez użytkownika, a niektóre systemy operacyjne wprowadzają jeszcze zabezpieczenie w postaci kosza. Jest on wydzielonym katalogiem, do którego trafiają skasowane pliki. Pliki umieszczone w koszu można jeszcze odzyskać, przenosząc je do innego wskazanego katalogu, a pozostałe należy kasować.

Porządkowanie i zabezpieczanie plików

W czasie użytkowania komputera w pamięci tworzy się coraz więcej katalogów, w których gromadzi się coraz więcej plików. Z czasem część zgromadzonych danych przestaje być aktualna bądź traci na znaczeniu, pojawia się potrzeba innego niż dotychczas ich uporządkowania. W przypadku braku systematyczności w gromadzeniu i porządkowaniu plików może powstać sytuacja, że użytkownik traci znajomość tego, co zgromadził, a odnalezienie potrzebnej informacji staje się uciążliwe. Nie ma ogólnych reguł postępowania w celu unikania tego rodzaju sytuacji. Dwie proste zasady, które z całą pewnością redukują ryzyko powstania takich sytuacji brzmią:

1. Używaj tylko tego oprogramowania, które znasz i którego potrzebujesz!

2. Przechowuj w pamięci tylko te dane, które znasz! Natomiast te, które znasz, ale których nie potrzebujesz w dającej przewidzieć się przyszłości, przechowuj w katalogach archiwalnych.

Pamięć należy chronić przed utratą zawartości. Przyczyny utraty mogą być różne, np. kradzież lub zniszczenie komputera czy trwała awaria dysku. W celu zapobieżenia skutkom takich sytuacji należy:

- sporządzać kopie systemu operacyjnego,
- okresowo archiwizować zawartość pamięci, a przynajmniej najbar dziej cennych plików na nośnikach zewnętrznych.

Mniejsze straty, np. utratę tych części plików, które były modyfikowane w okresie od ostatniej aktualizacji do chwili wystąpienia awarii, może powodować niespodziewana przerwa w zasilaniu. Straty te można znacznie ograniczyć przez okresowe składowanie (zapisywanie) plików roboczych.

Utrata zawartości pamięci może nastąpić w wyniku akcji nieupoważnionego użytkownika. Aby temu przeciwdziałać, należy dbać o staranny dobór haseł chroniących przed niepowołanym dostępem i odpowiednie przydzielanie praw dostępu, zwłaszcza do pamięci komputerów podłączonych do Internetu.

Ochrona przed wirusami

Bardzo duże zagrożenie powodują szeroko rozpowszechniające się wirusy. Są one przenoszone przez dyskietki, a w ostatnim okresie przez Internet, zwłaszcza przez pocztę elektroniczną. Trzeba mieć świadomość, że spotkanie z wirusem jest tylko kwestią czasu. Należy więc prowadzić odpowiednią profilaktykę, aby spotkanie to możliwie opóźnić, albo przynajmniej zminimalizować poniesione straty. W skrajnym przypadku utracić można całą zawartość pamięci komputera — oznacza to utratę nie tylko wszystkich plików z naszymi danymi, ale również konieczność ponownej instalacji systemu operacyjnego komputera oraz wszystkich aplikacji.

Wirus komputerowy to mały program, który ma następujące cechy:

- „przykleja się” do różnych programów użytkowych, to znaczy dołącza się do różnych plików; po stwierdzeniu, że znany nam rozmiar pliku został powiększony, możemy podejrzewać, że plik został zainfekowany;
- uruchomienie zainfekowanego programu ma zwykle dwa skutki: po pierwsze wirus powiela się i swoje kopie „dokleja” do innych programów użytkowych, a po drugie wykonuje czynności niebezpieczne dla danych, aplikacji i dla samego komputera.

Istnieje ogromna różnorodność wirusów. Wirusy, których działanie ogranicza się do powielania swoich kopii, nazywa się robakami. Niektóre wirusy swą destrukcyjną działalność uwidaczniają dopiero po upływie pewnego czasu od zainfekowania komputera, np. w określonym dniu kasują pewne pliki, zmieniają rozszerzenia ich nazw, a nawet całkowicie wyzerowują zawartość dysku — nazywa się je bombami logicznymi. Są też wirusy, które pozwalają przez Internet przejąć kontrolę nad zainfekowanym komputerem; jego właściciela może spotkać wszystko — od kradzieży da-

nych aż do skasowania zawartości dysku. Tego rodzaju wirusy, najczęściej przesyłane pocztą elektroniczną, sprawiają wrażenie pożytecznego programu, gdy tymczasem prowadzą destrukcyjną działalność — nazywa się je koniami trojańskimi.

Nie ma zasad gwarantujących bezwirusową pracę. Istnieją jednak reguły pozwalające zminimalizować ryzyko zarażenia, a w razie infekcji zmniejszyć skutki destrukcyjnej działalności wirusa.

1. Należy używać legalnie zakupionego oprogramowania. Minimalizuje to prawdopodobieństwo, że zdobyty program posiada wirusa. Firmy rozpowszechniające oprogramowanie prowadzą wszechstronne testy antywirusowe sprzedawanych produktów. Dotyczy to zarówno dystrybutorów, jak i wytwórców oprogramowania. Nie należy instalować oprogramowania pochodzącego z nieznanego źródła.

2. Należy bezwzględnie tworzyć kopie zapasowe posiadanego oprogramowania i efektów własnej pracy (archiwizacja). Umożliwia to odzyskanie danych, lub chociaż ich części, po zniszczeniu danych na dyskach roboczych.

3. Należy chronić kopie zapasowe. Ochrona powinna obejmować np. zablokowanie dyskietki przed zapisem. Takiej dyskietki nie wolno nikomu pożyczać, a przed udostępnieniem należy sporządzić jej kopię.

4. Podstawową cechą, która pomaga ustrzec się przed wirusami jest ostrożność. Należy skrupulatnie sprawdzać dyskietki pochodzące z niepewnego źródła, a także te, do których dostęp miały inne osoby i były wykorzystywane na innych komputerach. Należy ostrożnie podchodzić do programów-ciekawostek. Każdy z nich w trakcie interesującego pokazu może zniszczyć zawartość dysku, może też uczynić to dopiero po kilku miesiącach pracy. W szczególności ostrożność należy wykazywać wobec tajemniczych programów przychodzących jako załączniki poczty elektronicznej. Po odebraniu od nieznanego nadawcy listu, który w załączeniu dostarcza program, najlepiej od razu cały list skasować.

5. Konieczne jest posiadanie pewnej, zabezpieczonej dyskietki systemowej, zawierającej programy antywirusowe. Dobrym zwyczajem jest ich uruchamianie programu antywirusowego z zabezpieczonej dyskietki, gdyż w ten sposób eliminujemy możliwość zainfekowania programu antywirusowego, dla którego wirus mógłby stać się niewidoczny.

6. Należy wyposażyć oprogramowanie komputera w program antywirusowy. Program taki należy uruchamiać okresowo, zawsze przed zapisem do pamięci wewnętrznej z dyskietki, lub też uruchomić program na stałe. Stale uruchomiany program pracuje w tle, to znaczy nie jest bezpośrednio

widoczny przez użytkownika, a jego praca polega na stałym sprawdzaniu otwieranych i wykorzystywanych plików. Jest to zalecany tryb pracy programu, ale należy zdawać sobie sprawę, że jego wykonywanie może spowodować spowolnienie wykonywania aplikacji użytkowych.

7. Program antywirusowy umożliwia wykrycie i zniszczenia wirusa, a także często usunięcie skutków jego działania. Należy jednak pamiętać, że walka z wirusami jest nieustanna, ponieważ wciąż powstają nowe wirusy, a programy antywirusowe wymagają okresowej aktualizacji.

Programy antywirusowe są najskuteczniejszym narzędziem do zwalczania wirusów. Istniejące i ciągle rozwijające się programy ułatwiają ochronę systemu przed inwazją, szybkie jej wykrycie i w większości przypadków usunięcie skutków.

Podane niżej witryny internetowe dostarczają szczegółów na temat najbardziej popularnych programów antywirusowych oraz warunków ich zakupu i aktualizacji.

Mks_vir: www.mks.com.pl

Panda Antivirus: www.pandasoftware.com

Kaspersky Anti-Virus (AVP): www.kaspersky.com

Sophos Anti-virus: www.sophos.com

McAfee VirusScan: www.mcafee.com

AntiVirenKit: www.gdata.com.pl

Norton AntiVirus: www.symantec.com.pl

VI. SIECI KOMPUTEROWE

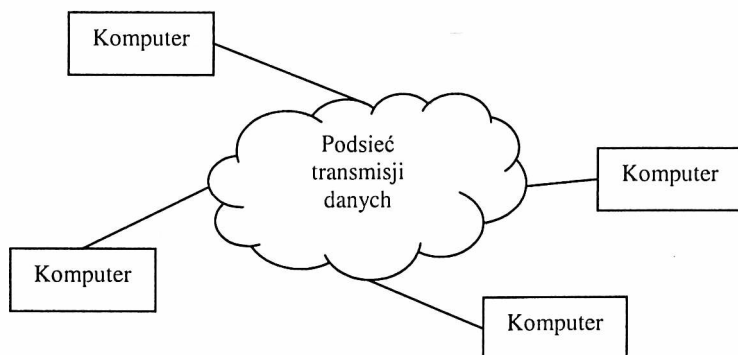
Co to jest sieć komputerowa?

Daty narodzin sieci komputerowych można doszukiwać się w końcu lat sześćdziesiątych, intensywny ich rozwój przypada na drugą połowę lat osiemdziesiątych, od początku lat dziewięćdziesiątych zaczęły się rozpowszechniać w Polsce, a obecnie na świecie i w Polsce stanowią standardowe narzędzie pracy i komunikacji.

Sieć komputerowa to zbiór komputerów i urządzeń zewnętrznych połączonych ze sobą za pośrednictwem podsieci transmisji danych, w celu wspólnej wymiany danych i współużytkowania zasobów — wspólny sprzęt, programy, bazy danych. Sieci umożliwiają wykorzystywanie tych samych zasobów informacji i urządzeń peryferyjnych przez wielu użytkowników, szybkie przesyłanie, wyszukiwanie i rozpowszechnianie informacji.

Rys. 6.1

Schematyczna struktura sieci komputerowej

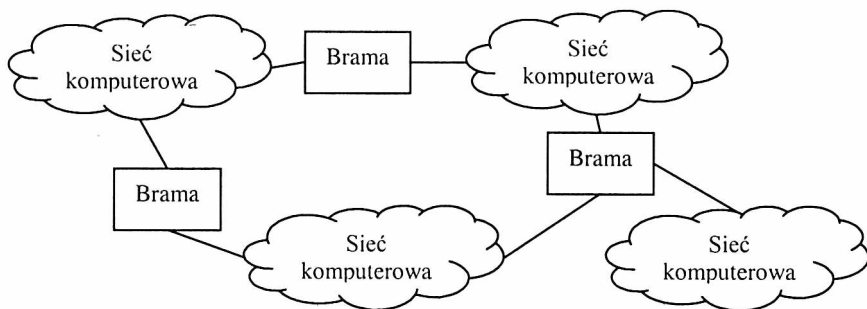


Usługi, z których korzysta użytkownik sieci, można sprowadzić do trzech zasadniczych kategorii:

- usługi informacyjne — polegają na dostępie do serwisów informacyjnych (np. encyklopedycznych, prawnych, prasowych), wyszukiwaniu, odczytaniu i sprowadzeniu do własnego komputera wybranych informacji;
- usługi komunikacyjne — służą do przekazywania pomiędzy użytkownikami sieci różnej postaci informacji: przesyłania listów elektronicznych (odpowiednik klasycznej poczty), prowadzenia pogawędek, przesyłania dokumentów (np. rozliczeń podatkowych, dokumentów bankowych);
- usługi obliczeniowe — pozwalają na wykorzystanie mocy obliczeniowej innych komputerów; użytkownik może przysyłać swoje dane albo swój program wraz z danymi w celu przeprowadzenia obliczeń na innym komputerze.

Podane wyżej określenie sieci komputerowej jest bardzo ogólne, gdyż odnosi się zarówno do sieci, jak i do federacji sieci, czyli sieci powiązanych ze sobą sieci. Właśnie z federacją wzajemnie powiązanych ze sobą sieci mamy obecnie często do czynienia, pracując w Internecie — największej, globalnej sieci komputerowej, obejmującej swoim zasięgiem cały świat.

Rys. 6.2
Schematyczne połączenie sieci



Federacja sieci powstaje przez połączenie sieci za pośrednictwem specjalnych komputerów — *bram* lub *bramek*, pełniących rolę pośrednika w wymianie danych pomiędzy sieciami. Ze względu na zasięg rozróżnia się kilka typów sieci komputerowych:

- domowe,
- lokalne,
- metropolitalne,
- rozległe.

Podział ten wynika zarówno ze względów technicznych, jak i organizacyjnych. Większy zasięg sieci wymaga bardziej rozbudowanej infrastruktury technicznej, a także — z uwagi na zaangażowanie większej liczby ludzi i instytucji — odpowiedniej organizacji, co pociąga za sobą różne problemy prawne, finansowe itp.

Sieciami o najmniejszym zasięgu są *sieci domowe*. Najnowsze rozwiązania technologiczne umożliwiają budowę bezprzewodowych sieci, które łączą rozmieszczone w mieszkaniu komputery i współpracujące z nimi urządzenia zewnętrzne. Zasięg takich sieci, mierzony odległością pomiędzy skrajnie położonymi elementami, jest rzędu kilkunastu metrów. Sieci te określa się także mianem *sieci prywatnych* — PAN (*Private Area Network*).

Większy zasięg, zwykle budynku lub kilku położonych obok siebie budynków, mają *sieci lokalne* — LAN (*Local Area Network*). *Sieci miejskie* (metropolitalne) — MAN (*Metropolitan Area Network*) są swym zasięgiem ograniczone do aglomeracji. Najszerszy zasięg, krajowy lub międzynarodowy, mają *sieci rozległe* — WAN (*Wide Area Network*).

Połączenie wielu sieci różnych rodzajów w jedną wspólną sieć nazywamy *internetem* (pisanym z małej litery), natomiast największy z internetów, czyli globalna federacja sieci o zasięgu światowym to *Internet* (pisany z dużej litery).

Obecnie w skład Internetu wchodzi kilka milionów sieci komputerowych o różnym charakterze — sieci akademickie, komercyjne, militarne, badawcze — łączące ludzi w niemal wszystkich krajach świata.

Dwoma podstawowymi składnikami każdego rodzaju sieci komputerowej są: podsieć transmisji danych (sieć telekomunikacyjna) oraz podłączone do niej komputery. Komputery te, ze względu na pełnione role w sieci, dzieli się na dwie kategorie: *serwery sieciowe* i *stacje robocze*.

Serwery sieciowe to komputery o dużej mocy obliczeniowej i dużych możliwościach gromadzenia oprogramowania i danych. Ich zadaniem jest udostępnianie gromadzonych danych oraz wykonywanie zainstalowanych aplikacji.

Stacje robocze to komputery o mniejszej mocy obliczeniowej, które pełnią rolę komputerów lokalnych i które pozwalają ich użytkownikom,

za pośrednictwem podsieci transmisji danych, sięgając do zasobów serwerów sieciowych.

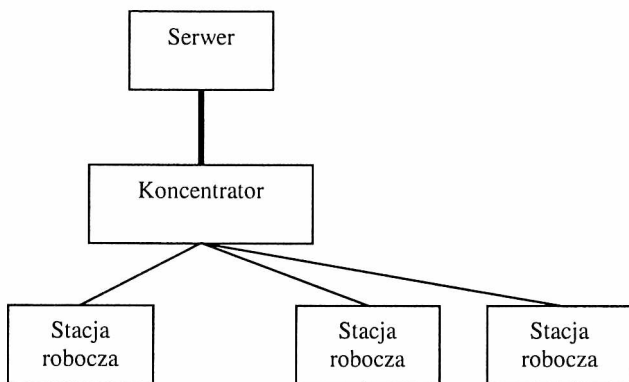
Są też oczywiście komputery, które mogą jednocześnie pełnić rolę serwera i stacji roboczej. Zarówno serwery, jak i stacje robocze posiadają odpowiednie oprogramowanie sieciowe, dzięki któremu mogą się wzajemnie komunikować.

Podsieć transmisji danych jest wyspecjalizowaną siecią komputerową, której jedyną usługą jest transmisja danych pomiędzy dołączonymi do niej komputerami czy systemami komputerowymi. Dlatego zamiast *podsieć* mówimy często *sieć transmisji danych*. W zależności od typu sieci są one różnie zbudowane. Podsieć transmisji jest zbiorem węzłów transmisyjnych połączonych łączami transmisyjnymi. Łączami są różnego rodzaju kable elektryczne (np. kable koncentryczne, pary skrętek), światłowody, kanały radiowe i satelitarne. Do przesyłania danych wykorzystuje się także linie energetyczne, a czasem istniejące systemy łączności telefonicznej.

Węzłami są urządzenia pełniące rolę koncentratorów (*hub*) lub/i przełączników (*router*). Rolą koncentratora jest pośredniczenie w transmisji pomiędzy dużą liczbą łączy o małej szybkości transmisji a zwykle jednym szybkim łączem transmisyjnym. Natomiast rolą przełącznika jest przekierowywanie transmisji z jednego łącza na inne.

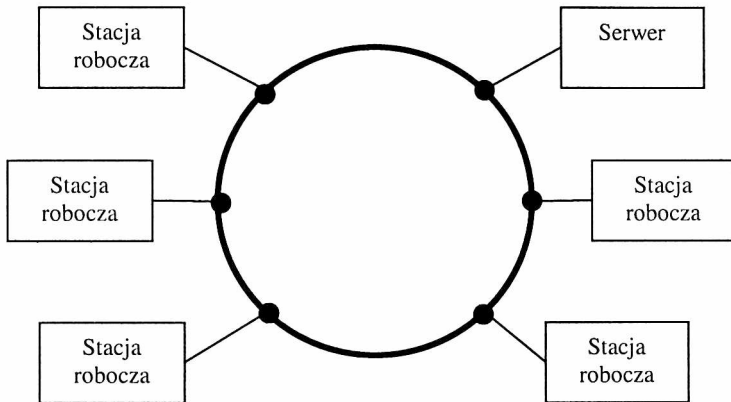
By zilustrować budowę podsieci transmisji danych, omówimy strukturę lokalnych sieci komputerowych. Najprostsza lokalna sieć komputerowa ma przynajmniej jeden serwer i kilka stacji roboczych. Mogą być one połączone ze sobą w strukturę: gwiazdową, pierścieniową lub szynową (magistralową).

Rys. 6.3

Sieć lokalna o strukturze gwiazdowej

W sieci lokalnej o strukturze gwiazdzystej serwery i stacje robocze są przyłączone do jednego wspólnego koncentratora. Taka struktura i odpowiednia konstrukcja koncentratora umożliwiają serwerom i stacjom roboczym jednoczesny dostęp do łącz transmisyjnych i niezależne prowadzenie transmisji danych. Równoczesne prowadzenie transmisji z wielu stacji roboczych z jednym serwerem wymaga, by łącze transmisyjne pomiędzy koncentratorom a serwerem było wielokrotnie szybsze. Typowe zestawy szybkości transmisji to 10-100 Mb/s pomiędzy koncentratorom a serwerem i 1-10 Mb/s pomiędzy koncentratorom a stacjami roboczymi.

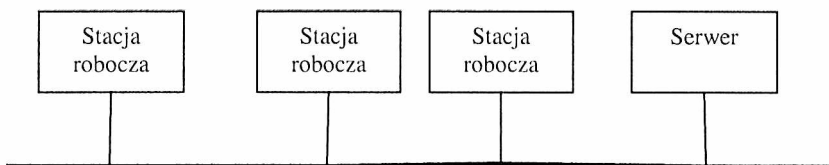
Rys. 6.4
Sieć lokalna o strukturze pierścieniowej



W sieci o strukturze pierścieniowej łącze transmisyjne tworzy zamkniętą pętlę — pierścień, do której są przyłączone serwery i stacje robocze. Organizacja transmisji danych jest tu bardziej złożona, ponieważ wszystkie stacje i serwery mają dostęp tylko do jednego wspólnego łącza. Dane są przesyłane zawsze w jednym kierunku od danej stacji do stacji sąsiedniej. Stacja, która odbiera adresowane do niej dane, zatrzymuje je dla siebie, a w odwrotnym przypadku przekazuje je dalej do swojego sąsiada. Przyjmuje się, że w danym momencie prawo do wysyłania danych ma tylko jedna stacja, a pozostałe mogą tylko odbierać dane. Stacja, która ma prawo nadawania danych, może z tego prawa skorzystać, a po wysłaniu danych lub w przypadku braku takiej potrzeby przekazuje prawo do wysyłania danych do innej, np. sąsiedniej stacji. W zależności od potrzeb stosuje się różne algorytmy przekazywania uprawnień do trans-

misji pomiędzy węzłami sieci. Sieci o tej strukturze są wykorzystywane w zastosowaniach przemysłowych, np. w sterowaniu liniami technologicznymi. Stosowane szybkości transmisji są bardzo różne, typowy przedział szybkości wynosi od 9,6 Kb/s do 12 Mb/s.

Rys. 6.5

Sieć lokalna o strukturze magistralowej

Strukturę magistralową wyróżnia to, że wszystkie węzły sieci połączone są ze sobą za pomocą pojedynczego, otwartego (umożliwiającego przyłączenie kolejnych urządzeń) łącza transmisyjnego noszącego nazwę magistrali. Sieci o strukturze magistralowej należą do najbardziej popularnych sieci lokalnych w zastosowaniach biurowych, spotykane szybkości transmisji wynoszą od 1 Mb/s do 1 Gb/s.

Ciekawy jest tu sposób organizacji transmisji danych pomiędzy węzłami sieci, oparty na tzw. protokole rywalizacyjnym. Ze względu na dostęp do wspólnego łącza — tak jak w przypadku sieci pierścieniowych — możliwe jest w danym momencie wysyłanie danych tylko przez jedną stację. Nie ma tu jednak wyróżnionej stacji mającej prawo nadawania, może to czynić każda. Jeżeli jednak zajdzie taka sytuacja, że jednocześnie czy prawie jednocześnie zaczną nadawać dwie lub więcej stacji, to każda z nich — nadając, ale jednocześnie i nasłuchując sygnału w łączy transmisyjnym — stwierdzi kolizję nadawania. W takim przypadku każda z nadających stacji ma obowiązek przerywania nadawania i może je wznowić dopiero po upływie pewnego czasu. Aby jednak przy próbie wznowienia nadawania nie nastąpiła kolejna kolizja, czas, po którym następuje wznowienie nadawania, jest przez każdą stację ustalany w sposób losowy. Rozkłady prawdopodobieństwa dobiera się tak, aby zminimalizować średni czas oczekiwania na wznowienie nadawania. Jeśli powtórne wznowienie zakończy się kolizją, nadające stacje postępują jak przy pierwszej kolizji.

Budowa sieci transmisji danych, obsługującej rozległą sieć komputerową, jest o wiele bardziej złożona, przypomina nieco światową sieć łączności telefonicznej. W obrębie takiej sieci transmisji danych można wy-

różnić przynajmniej trzy rodzaje sieci transmisyjnych różniących się parametrami technicznymi, a przede wszystkim szybkością transmisji:

- szkieletowe — obsługujące ruch międzynarodowy, czasem między-miastowy, transmitujące dane z szybkością setek Gb/s,
- regionalne (pośredniczące) — obsługujące ruch międzymiastowy i regionalny, z szybkością pojedynczych Gb/s,
- miejskie, ograniczone do aglomeracji — transmitujące z szybkością kilkudziesięciu-kilkuset Mb/s.

Podane szybkości transmisji są orientacyjne, gdyż obserwuje się trwałą tendencję do ich zwiększania.

Identyfikacja komputerów

Jak jest możliwe funkcjonowanie sieci, która jest ogromnym zbiorem komputerów zróżnicowanych sprzętowo i programowo? — oto zasadnicze pytanie, jakie się nasuwa przy wyjaśnianiu działania sieci.

Pierwszym szczegółowym problemem jest adresowanie stacji roboczych i serwerów. Sposób adresowania komputerów pracujących w Internecie odzwierciedla podział całej sieci na tzw. domeny, który leży w gestii Departamentu Obrony USA. Wynika to z faktu, że Internet powstał na bazie projektów sieci badawczych finansowanych przez Departament Obrony. Domeny te mogą być podzielone na wewnętrzne poddomeny. Domeny internetowe najwyższego poziomu są podzielone ze względu na położenie geograficzne i charakter organizacyjny. Domeny niższych poziomów są wewnętrznie podzielone na podstawie kryteriów ustanowionych przez ich administratorów.

Każdy komputer podłączony do Internetu ma unikalny adres (identyfikator), tzw. adres TCP/IP (*Transmission Control/Internet Protocol*). Składa się on z 4 części, które są liczbami całkowitymi z zakresu od 0 do 255. Wynika to z tego, że na każdą część adresu przeznaczono w pamięci komputera 1 bajt, a na cały adres — 4 bajty. Liczba możliwych adresów wynosi ok. 4,2 mld, co wkrótce może być istotnym ograniczeniem rozwoju Internetu.

Adres zapisuje się w postaci ciągu liczb dziesiętnych oddzielonych kropkami, np.:

123.32.67.20

Adresy sieci przydzielane są przez organizacje zarządzające Internetem. Aby połączyć się z wybranym serwerem, trzeba znać jego adres, podobnie jak przy połączeniach telefonicznych numer telefonu.

Ponieważ zapamiętanie adresu serwera jest kłopotliwe, przyporządkowano im symboliczne nazwy, zwykle w postaci:

`nazwaKomputer.nazwaPoddomeny2.nazwaPoddomeny1.domena`

Np. nazwa

`zh-p5.ci.pwr.wroc.pl`

ma adres

`156.17.131.11`

Nazwy te, określane także mianem adresów domenowych, odzwierciedlają hierarchiczną strukturę podziału Internetu na domeny. Ostatni człon nazwy `pl` jest nazwą domeny najwyższego poziomu, tu oznaczającą nazwę kraju (Polska), której odpowiada numer 156, `wroc` jest nazwą poddomeny — odpowiednik numeru 17, `pwr` jest nazwą kolejnej poddomeny — odpowiednik numeru 131, człon `zh-p5.ci` jest nazwą komputera — odpowiednik numeru 11.

Domeny internetowe odnoszące się do innych wybranych krajów mają postać:

`us` — Stany Zjednoczone,

`uk` — Wielka Brytania,

`de` — Niemcy,

`ru` — Rosja,

`fr` — Francja.

Domeny internetowe odnoszące się do wybranych organizacji mają postać:

`com` — organizacje handlowe,

`edu` — uniwersytety i inne instytucje edukacyjne,

`gov` — przedstawicielstwa rządowe,

`mil` — organizacje militarne,

`net` — ważniejsze centra wspomagające działanie sieci,

`int` — organizacje międzynarodowe,

`org` — inne organizacje.

Domeny nie zawsze precyzyjnie określają w nazwie charakter serwera, dlatego niełatwo stwierdzić, gdzie fizycznie znajduje się komputer o podanej nazwie. Dodatkowym utrudnieniem może być to, że jeden komputer może mieć kilka adresów.

Informacje o adresach i przyporządkowanych im nazwach umieszczone są na specjalnych serwerach nazwanych DNS (*Domain Name System*). Każda domena posiada własny serwis adresowy. Gdy użytkownik podaje nazwę, jego komputer łączy się z najbliższym serwerem DNS, który podaje odpowiadający jej adres, przy czym w przypadku stwierdzenia, że nazwa nie należy do danej domeny, serwer komunikuje się z serwerem DNS domeny wyższego poziomu.

Z punktu widzenia użytkownika adres sieciowy jego komputera nie jest najważniejszy. Z adresem spotyka się on w zasadzie tylko podczas podłączania komputera do Internetu, podobnie jak z adresem serwera, do którego komputer ma bezpośredni dostęp. Z praktycznego punktu widzenia dla użytkownika ważniejszy jest adres jego skrzynki pocztowej oraz adresy zasobów informacyjnych, z których chciałby korzystać.

Sieci otwarte

Kolejne problemy, które wiążą się z funkcjonowaniem sieci, dotyczą formatów oraz znaczenia wymienianych danych.

Np. by tekst opracowany w jednym kraju i przesłany do innego kraju mógł być odczytany, konieczna jest nie tylko znajomość formatów zapisów tekstowych stosowanych w obu krajach, ale również programy transformacji tych formatów. Okazuje się, że w wielu przypadkach taka transformacja jest niejednoznaczna, a w skrajnych — nie jest nawet możliwa. Podobne problemy wiążą się z przesyłaniem i interpretacją plików medialnych.

Jeszcze trudniejszym problemem byłoby przetłumaczenie tekstu napisanego w jednym języku na inny język. Rozwiązanie tego problemu nie jest wprawdzie konieczne do poprawnego funkcjonowania sieci, ale stanowi dobrą ilustrację trudności, jakie niesie współpraca komputerów w Internecie.

W celu umożliwienia współpracy w jednej sieci komputerów różniących się sprzętowo i programowo opracowano standardy współpracy. Te systemy komputerowe, które spełniają standardy, nazywa się systemami otwartymi. W ramach Internetu opracowano zestaw standardów, których ogólna idea polega, podobnie jak w przypadku oprogramowania komputera osobistego, na wprowadzeniu warstw funkcjonalnych. Warstwy te są uporządkowane hierarchicznie: dana warstwa dostarcza pewnych usług warstwie wyższej i w celu realizacji tych usług korzysta z usług warstwy niższej. Standaryzacja polega na ustaleniu *hierarchii warstw*, określeniu

usług, jakie dana warstwa oferuje, i sposobu ich realizacji za pomocą usług warstwy niższej, czyli *protokołu* warstwy.

Upraszczając, całą hierarchię można sprowadzić do dwóch warstw: warstwy transmisji danych i warstwy aplikacji. Warstwa transmisji danych (warstwa transportowa), obejmując niższe warstwy hierarchii, świadczy usługi wyższym warstwom, z których najwyższą jest warstwa aplikacji świadcząca usługi użytkownikom sieci.

W Internecie do warstwy transmisji danych odnosi się zestaw protokołów określanych skrótem TCP/IP. Warstwa ta świadczy zasadniczo dwie usługi:

- usługa transmisji połączeniowej — polega na utworzeniu połączenia transmisyjnego i niezawodnego przesyłania danych pomiędzy dwoma komputerami; można ją porównać do usługi połączenia telefonicznego; po nawiązaniu połączenia pomiędzy dwoma abonentami mogą oni prowadzić jednoczesną wymianę wiadomości;
- usługa transmisji bezpołączeniowej — polega na niezawodnym przesłaniu pojedynczej wiadomości od jednego komputera do drugiego bez uprzedniego tworzenia połączenia; można ją porównać do usługi przesłania wiadomości SMS w telefonie komórkowym; wariantem tej usługi jest *rozgłaszanie*, czyli jednoczesne wysyłanie tej samej wiadomości od jednego nadawcy do wielu odbiorców.

Na bazie usług warstwy transmisyjnej funkcjonuje warstwa aplikacji sieciowych, która dostarcza licznych usług, dostępnych bezpośrednio użytkownikom sieci. Do najbardziej popularnych należą usługi:

- WWW (*World Wide Web*),
- poczty elektronicznej (*e-mail*),
- transferu plików (*File Transfer Protocol*),
- zdalnego dostępu (*Telnet*),
- zdalnych pogawędek IRC (*Internet Relay Chat*).

Światowa pajęczyna WWW

Nazwa usługi — *World Wide Web*, czyli światowa pajęczyna — dobrze oddaje jej zasięg i znaczenie. Obecnie jest to najpopularniejsza usługa w Internecie. Jej specyfika opiera się na pojęciu *dokumentu hipertekstowego* — uogólnieniu dokumentu tekstowego, oraz na powiązaniach pomiędzy dokumentami hipertekstowymi.

Dokument hipertekstowy, oglądany na ekranie monitora, różni się od zwykłego dokumentu tekstowego lub tekstowo-graficznego tym, że jego wyróżnione wizualnie (zwykle przez podkreślenie, ale także przez inny krój pisma lub kolor) elementy składowe pełnią funkcję odsyłaczy do innych fragmentów tego samego dokumentu albo do innych dokumentów. Odsyłacz, inaczej łącznik hipertekstowy lub odnośnik (*link*), proponuje użytkownikowi przejście w inne miejsce tego samego lub innego dokumentu.

Czytelnik może z tej propozycji skorzystać, co w przypadku interfejsu graficznego polega na wskazaniu kursorem wybranego łącznika i kliknięciu myszą. Czynność ta uruchamia akcję, której efektem jest wyświetlenie na ekranie odpowiedniego fragmentu tego samego lub innego dokumentu. Jeśli odsyłacz wskazuje na dokument znajdujący się w innym komputerze, to w ramach takiej akcji wykonywanych jest wiele czynności: komputer, na którym jest czytany dokument hipertekstowy, musi zidentyfikować inny komputer, który przechowuje dokument wskazywany przez odsyłacz, nawiązać z nim połączenie w celu sprowadzenia tego dokumentu i dopiero po jego odbiorze wyświetlić go na ekranie.

Sposób nawiązywania połączenia i przesyłania żądanych dokumentów pomiędzy komputerami jest zestandaryzowany — podstawowym jest *protokół http* (*Hypertext Transfer Protocol*), służący do przesyłania dokumentów hipertekstowych zapisywanych w standardzie HTML (*Hypertext Markup Language*). Język HTML służy do definiowania dokumentów hipertekstowych, obecnie jest to ciągle najbardziej popularny język, chociaż zaczyna już być zastępowany przez język XML (*eXtended Markup Language*).

Wybierane dokumenty hipertekstowe są zawartością odpowiednich plików. Odwoływanie się do wskazanego dokumentu oznacza wyszukanie i sięgnięcie do odpowiedniego pliku znajdującego się w pewnym komputerze. Jak omawialiśmy wcześniej, każdy komputer ma swój adres internetowy, a każdy plik w pamięci danego komputera jest jednoznacznie identyfikowany przez ścieżkę dostępu. Zatem z każdym odnośnikiem jest skojarzony identyfikator (*adres*) zasobu sieciowego URI (*Universal Resource Identifier*). Identyfikator taki składa się z trzech części:

- symbolu protokołu udostępniania danych; w przypadku protokołu http jest to `http://`;
- adresu komputera w postaci symbolicznej lub numerycznej;
- ścieżki dostępu pliku w pamięci komputera.

Np. identyfikatorem zasobu sieciowego jest:

<http://wiem.onet.pl/wiem/0137cb.html>

Fragment wiem.onet.pl/wiem jest sieciowym adresem polskiego serwera, który udostępnia informacje encyklopedyczne, a [0137cb.html](http://wiem.onet.pl/wiem/0137cb.html) jest nazwą pliku hipertekstowego zawierającego wyjaśnienie hasła *hipertekst*.

Do wykonywania czynności wyświetlania dokumentów hipertekstowych, przechodzenia do innych fragmentów zgodnie z wybieranymi odnośnikami potrzebne jest odpowiednie oprogramowanie zainstalowane w obu komunikujących się komputerach.

Komputer, który udostępnia swoje dokumenty — serwer WWW — musi je odpowiednio przygotować w postaci *stron WWW* (lub HTML), liczba takich stron udostępnianych obecnie w Internecie wynosi setki milionów i nieustannie rośnie.

Komputer, który dokumenty te ogląda — klient WWW — powinien mieć zainstalowaną *przeglądarkę dokumentów WWW*. Do najpopularniejszych przeglądarek dostępnych na rynku należą: *Netscape Navigator* i *Internet Explorer*. Przeglądarki stają się obecnie standardową częścią systemów operacyjnych, co oznacza, że kupując nowy komputer wraz z systemem operacyjnym, mamy już możliwość dostępu do internetowych stron WWW. Przeglądarki oprócz wymienionych wykonują jeszcze inne czynności, m.in. zapamiętują ciąg odwiedzanych stron WWW, co pozwala użytkownikowi na prosty powrót ze strony, którą aktualnie ogląda na stronę, z której rozpoczął wędrowkę, dodatkowo użytkownik może stworzyć własny katalog ulubionych stron, do których zamierza sięgać.

Dysponując przeglądarką, użytkownik przez oglądanie kolejnych dokumentów może wędrować po internetowych stronach całego świata. Wędrowka taka może być przypadkowa w tym sensie, że polega tylko na przechodzeniu ze strony na stronę przy wykorzystywaniu odsyłaczy w oglądanych dokumentach. O wiele częstsza jest potrzeba znalezienia informacji na konkretny temat w sytuacji, gdy użytkownik nie dysponuje adresem internetowym potrzebnych dokumentów. Do odnajdywania stron WWW z informacjami na dany temat służą specjalne serwisy wyszukiwawcze — *wyszukiwarki*. Podobnie jak inne zasoby sieciowe, mają one swoje adresy internetowe. Liczba dostępnych wyszukiwarek jest dość duża, rzędu kilkudziesięciu. Poniżej przedstawiono adresy kilku popularnych wyszukiwarek:

AltaVista: <http://www.altavista.com>

Google: <http://www.google.pl>

Infoseek: <http://infoseek.icm.edu.pl>

Lycos: <http://www.lycos.com>

Yahoo: <http://www.yahoo.com>

WebCrawler: <http://webcrawler.com>

Wyszukiwanie potrzebnych dokumentów za pomocą wyszukiwarek polega na wykonaniu następujących czynności. Po pierwsze, korzystając z własnej przeglądarki, należy nawiązać połączenie z wyszukiwarką. Polega to na wpisaniu adresu sieciowego wyszukiwarki w odpowiednie okienka przeglądarki i poleceniu nawiązania połączenia (naciśnięcie klawisza akceptacji lub kliknięcie myszą). Po uzyskaniu połączenia z wyszukiwarką należy sformułować pytanie. Pytanie jest zestawem słów charakteryzującym daną dziedzinę, a odpowiedzią jest pewna ilość adresów tych dokumentów, które mają związek z pytaniem, tj. ich treść dotyczy wskazanej dziedziny.

Wyszukiwarki oferują różne sposoby ustalania związku pomiędzy pytaniem a odpowiedzią. Np. jeśli interesuje nas informacja na temat rozgrywek piłki nożnej, to jako pytanie możemy wprowadzić zestaw słów „w piłkę nożną”. Możemy określić, czy chodzi nam o dokumenty, które w swej treści zawierają pełny zwrot „w piłkę nożną”, czy o dokumenty, które zawierają wszystkie słowa, czy też zawierają przynajmniej jedno ze słów z podanego zestawu. Różne wyszukiwarki mają różne mechanizmy dotyczące tych ustaleń, co nie stanowi jednak trudności, gdyż udostępniają one odpowiednie instrukcje na temat swego działania.

Formułowanie pytania na dany temat wymaga pewnej staranności. Przy zbyt ogólnej charakterystyce dziedziny otrzymamy w odpowiedzi dużą ilość stron, spośród których tylko nieliczne będą zawierać to, co nas interesuje. Natomiast przy zbyt wąskiej charakterystyce zbiór stron, który otrzymamy jako odpowiedź, może nie zawierać naprawdę interesujących nas stron.

Ilustracją problemu jest poprzednio omawiany przykład, gdy interesują nas rozgrywki w piłkę nożną. Zamiast poprzedniego zestawu słów „w piłkę nożną” lepszym wydaje się zestaw „rozgrywki w piłkę nożną”. Potwierdza to eksperyment z wyszukiwarką Google, która odpowiadając na pytanie z pierwszym zestawem słów, odnalazła ponad 8 tys. adresów, na pytanie z drugim zestawem słów — ok. 740 adresów, a na zestaw „ligowe rozgrywki w piłkę nożną” — 72 adresy.

W drodze ewolucji wyszukiwarki przekształciły się w *portale internetowe*, w których funkcja wyszukiwania informacji jest tylko jedną z wielu proponowanych usług. Oprócz niej portale oferują serwisy informacyjne zawierające katalogi uporządkowane według pewnych kryteriów. Mogą to być katalogi poświęcone np. sportowi, nauce, polityce, rozrywce, nowościom kinowym. Zawierają one zestaw odsyłaczy do najważniejszych stron z danej dziedziny. Jeśli poszukuje się typowych informacji na dany temat, są bardzo pomocne i pozwalają szybciej odnaleźć informację niż wyszukiwarki. Ponadto portale oferują darmowe konta poczty elektronicznej, grupy dyskusyjne, zakupy przez Internet.

Przykładem znanych polskich portali internetowych o charakterze ogólnym są:

Wirtualna Polska: <http://www.wp.pl>

Onet.pl: <http://www.onet.pl>

Polska.pl: <http://www.polska.pl>

Przykładem specjalizowanych polskich portali internetowych są:

Wrocławskie Centrum Sieciowo-Superkomputerowe:

<http://www.wcss.wroc.pl>

Polski Serwer Prawa: <http://www.prawo.lex.pl>

CBOS: <http://www.cbos.com.pl>

Polskie Towarzystwo Informatyczne: <http://www.pti.org.pl>

Telekomunikacja Polska SA: <http://www.tpsa.pl>

Należy zwrócić uwagę, że symboliczne adresy ogólnych portali internetowych, zwłaszcza wyszukiwarek, nie określają jednoznacznie adresu numerycznego serwera, na którym jest zainstalowana wyszukiwarka. Wynika to z faktu, że z daną wyszukiwarką jest związanych wiele serwerów, ulokowanych w różnych miejscach świata. Serwery te ze sobą współpracują, co nie jest widoczne dla użytkownika, tworząc system wyszukiwarki. Żądanie połączenia z wyszukiwarką jest kierowane — na podstawie informacji przekazanej z domenowego serwera nazw — do najbliższego serwera należącego do systemu danej wyszukiwarki.

Usługi poczty elektronicznej

Koncepcja usługi poczty elektronicznej (*e-mail*) jest oczywista — stanowi elektroniczną formę tradycyjnej poczty: zamiast listów papierowych korespondenci przesyłają pliki, których główną zawartością są teksty. W elektronicznej wersji zawartość poczty może być bogatsza, ponieważ

oprócz tekstów można przysyłać, zwykle w postaci załączników, niewielkie pliki z zawartością o dowolnym charakterze — z grafiką, dźwiękiem i obrazem ruchomym. Dodatkowo elektroniczna poczta umożliwia:

- edycję listów,
- definiowanie pilności wysyłanych listów,
- żądanie potwierdzenia odbioru listu przez odbiorcę,
- przekazywanie odbieranych listów do innych odbiorców,
- wysyłanie listu jednocześnie do wielu odbiorców,
- tworzenie katalogów służących zapamiętywaniu wysyłanych i odbieranych listów,
- tworzenie list adresowych indywidualnych i zbiorowych odbiorców wysyłanej korespondencji.

Nie są to wszystkie możliwości poczty, a dokładniej programów obsługujących pocztę elektroniczną. Podobnie jak przeglądarki, również programy obsługi poczty stają się integralną częścią systemów operacyjnych. Przykładem takich programów są *Outlook* oraz jego rozbudowana wersja *Outlook Express* w systemach *Microsoft Windows*.

Aby skorzystać z usługi poczty elektronicznej, trzeba mieć:

- fizyczny dostęp do sieci Internet,
- założone konto pocztowe na serwerze pocztowym (może to być serwer lokalnej lub osiedlowej sieci komputerowej lub serwer portalu informacyjnego, np. *Wirtualna Polska*, *Onet*),
- program pocztowy (np. *Outlook Express*, *Netscape Mail*) z prawidłowo wpisanymi do niego informacjami o naszym koncie pocztowym lub korzystać z przeglądarki WWW.

Użytkownicy poczty elektronicznej mają swoje adresy. Typowa struktura adresu wygląda następująco:

imię.nazwisko@serwerPocztowy.organizacja.kraj

Np.:

jan.kowalski@wszib.wroc.pl

Symbol @ stanowi skrót łacińskiego słowa *at*, w języku polskim należy go czytać *przy*.

Budowa listu elektronicznego różni się nieznacznie od budowy zwykłego listu. List elektroniczny składa się z dwóch części: nagłówka i właściwej treści. Nagłówek zawiera informacje wykorzystywane przez serwery pocztowe i programy pocztowe oraz ogólne dane o przesyłce, takie jak:

- nazwa adresata i jego adres,
- data wysłania przesyłki,
- nazwa nadawcy i jego adres,
- temat wiadomości.

Data wysłania i adres nadawcy są wpisywane automatycznie.

Szczególną usługą poczty elektronicznej jest *lista dyskusyjna*, która polega na przysyłaniu listów pomiędzy grupą użytkowników poczty — każdy list wysłany przez dowolnego członka grupy dyskusyjnej trafia do wszystkich pozostałych członków grupy. Istotą znaczenia usługi nie jest mechanizm automatycznego rozsyłania listów, ale możliwość efektywnego prowadzenia dyskusji na ustalony przez członków grupy temat. Efektywność zależy jednak od wewnętrznej organizacji i zachowania członków grupy. Sprawnie funkcjonująca lista dyskusyjna wymaga moderatora grupy, który stawia właściwe pytania i podsumowuje fragmenty dyskusji, a także od członków grupy, którzy powinni wypowiadać się na temat możliwie jasno i zwięźle oraz powstrzymywać się od nieistotnych wypowiedzi — od uczestników listy dyskusyjnej oczekuje się zachowania *netykiety* (termin ten pochodzi od słów: *net* — sieć i *etykieta*).

Inne usługi

Omówione wyżej usługi WWW i poczty elektronicznej są wykorzystywane najszerzej. Wynika to z ich uniwersalności — można nimi w znacznym stopniu zastąpić pozostałe usługi.

Usługa transferu plików (*File Transfer Protocol*) umożliwia użytkownikom dostęp i transfer plików zapisanych w zdalnych komputerach. Realizacja usługi polega na nawiązaniu połączenia z wybranym komputerem i — pod warunkiem posiadania odpowiednich uprawnień — wskazanie pliku do przesłania do własnego komputera. Często spotykamy sytuację, że nie jest konieczne posiadanie odpowiednich uprawnień, gdy zdalny komputer udostępnia pewne kartoteki plików użytkownikom anonimowym.

Usługa zdalnego dostępu (*Telnet*) polega na tym, że użytkownik stacji roboczej poprzez sieć loguje się w zdalnym komputerze podobnie jak w swoim własnym. Logowanie wymaga oczywiście uprzedniej rejestracji użytkownika w zdalnym komputerze, a więc nadania mu nazwy i ustalenia hasła. Po zalogowaniu użytkownik może pracować na zdalnym komputerze tak samo jak na komputerze lokalnym. Przy dostatecznie szyb-

kich łączach transmisyjnych opóźnienia w reakcji zdalnego komputera na kierowane do niego polecenia mogą być niezauważalne. Przykładem naturalnej potrzeby wykorzystania zdalnego dostępu jest sytuacja, gdy użytkownik danego serwera w sieci lokalnej znajdzie się w innym mieście i chce skorzystać z usług własnego serwera.

Z obiema usługami często wiąże się pewna niedogodność wynikająca z tego, że wymagana jest znajomość odpowiednich poleceń w postaci tekstowej. Może to być dość uciążliwe dla użytkownika przyzwyczajonego do pracy w interfejsie graficznym.

Usługa zdalnych pogawędek IRC (*Internet Relay Chat*) polega na bezpośredniej komunikacji pomiędzy różnymi użytkownikami sieci. Przesyłane komunikaty mają postać krótkich tekstów, które po napisaniu i zatwierdzeniu przez nadawcę prawie natychmiast trafiają do odbiorcy. Przypomina to interaktywną wymianę SMS-ów za pomocą telefonów komórkowych. Usługa ma kilka wariantów; jeden z wariantów polega na tym, że użytkownik ma na ekranie swego monitora dwa okienka, jedno służy mu do umieszczania własnych tekstów, a w drugim obserwuje powstawanie tekstów pisanych przez jego interlokutora.

VII. PODSTAWOWE ZASADY PRACY W SIECI KOMPUTEROWEJ

Bezpieczeństwo systemów komputerowych

Bezpieczeństwo systemu komputerowego wiąże się z nienaruszalnością sprzętu oraz pamiętanych danych. Pojęcie *bezpieczeństwa* ma szerszy zakres znaczeniowy od pojęcia *ochrony*, które związane jest z kontrolą dostępu użytkowników do zasobów systemu komputerowego.

Miarą bezpieczeństwa jest zaufanie użytkownika, że system i jego dane pozostaną nienaruszone. Określenie, które odwołuje się do pojęcia zaufania użytkownika, nie może być precyzyjne. Przy ocenie bezpieczeństwa systemów nie odwołujemy się do pojęcia zaufania, oceniamy bezpieczeństwo systemów pośrednio przez wyliczenie mechanizmów zabezpieczających. Dzięki temu, oceniając bezpieczeństwo dwóch systemów, możemy mówić, że bardziej bezpieczny jest ten, który posiada więcej mechanizmów zabezpieczających.

Dla zrozumienia zakresu mechanizmów zabezpieczających warto przedstawić rodzaje zagrożeń, na które narażony jest system komputerowy. Chodzi tu o zewnętrzne źródła zagrożeń związane z przypadkowymi lub świadomymi *atakami na bezpieczeństwo systemu*.

Pomijamy dalej środowiskowe źródła zagrożeń, takie jak ogień, dym, kurz, eksplozje, wyładowania atmosferyczne, wibracje, trzęsienia ziemi. Głównym zagrożeniem dla bezpieczeństwa pozostaje człowiek, którego działania mogą mieć charakter ataku *sprzętowego* lub *programowego*.

Najbardziej oczywistą formą ataku sprzętowego jest zniszczenie systemu lub linii telekomunikacyjnych. Z tego rodzaju zagrożeniem muszą się liczyć bardziej instytucje niż użytkownicy komputerów osobistych, którzy powinni bardziej obawiać się kradzieży lub pożaru.

Wyróżnia się *pasywne* i *aktywne* ataki programowe. Atak pasywny wiąże się z podglądaniem lub podsłuchem w celu uzyskania dostępu do zasobów przez osobę postronną. Mając możliwość odczytu danych, oso-

ba postronna może wykorzystywać poufne informacje, np. bankowe, handlowe, może też nielegalnie kopiować oprogramowania. W przypadku ataku aktywnego, czyli możliwości zmiany danych, straty mogą być jeszcze większe, chociaż na ogół wcześniej zauważane. Zmiana może polegać np. na zmianie stanu konta bankowego, zablokowaniu komputera przez upoważnionych użytkowników i przejęciu przez postronnego użytkownika roli administratora systemu. Atak aktywny może mieć różne formy, np.:

- maskarady, czyli podszywania się pod upoważnionego użytkownika w celu przechwycenia jego hasła,
- zgadywania haseł,
- modyfikowania przesyłanych komunikatów,
- blokowania działania systemu lub jego fragmentów,
- włamania do systemu przez wykorzystywanie nieszczelności systemu operacyjnego.

Ostatni rodzaj ataków wiąże się z działaniem hakerów (*hackers*). Początkowo przez hakerów rozumiano osoby, które ze względów ambicio-nalnych — pokazania swych umiejętności informatycznych — dokonywały włamań do systemów komputerowych, bez naruszania jego zasobów. Odróżniano ich w ten sposób od krakerów (*crackers*), czyli bezwzględnych włamywaczy-wandali, niszczących zasoby systemu. Obecnie oba terminy mają jednakowo negatywne, kryminalne konotacje.

Zasadniczymi powodami komputerowej przestępczości są:

- szpiegostwo przemysłowe, dotyczące zwłaszcza firm prowadzących badania nad nowymi technologiami lub konstrukcjami,
- chęć osiągnięcia korzyści materialnych przez manipulację na kon-tach bankowych, a także na operacjach całego systemu elektronicznego obrotu bezgotówkowego,
- pragnienie odwetu sfrustrowanych pracowników lub byłych pracowników firm i instytucji,
- pragnienie rozgłosu, związane najczęściej z pokazaniem własnej nieuchwytności.

W celu systematycznego zabezpieczenia systemów komputerowych w połowie lat osiemdziesiątych Departament Obrony USA opracował pierwszą klasyfikację poziomów bezpieczeństwa. Podobne opracowanie powstało na początku lat dziewięćdziesiątych w krajach Wspólnoty Europejskiej. Wyróżnia ono mechanizmy ochrony bezpieczeństwa, wśród których są m.in.:

- identyfikacja i uwierzytelnianie użytkowników — każdy użytkownik ma unikalny identyfikator (nazwę konta) i hasło, stanowiące potwierdzenie jego tożsamości;
- kontrola dostępu do zasobów — pozwala na udzielenie użytkownikowi dostępu tylko do tej części zasobów systemu, która jest mu niezbędna do jego pracy;
- prowadzenie dzienników pracy — pozwala na automatyczną rejestrację wybranych czynności wykonywanych przez użytkownika w celu możliwości rozliczenia wykonywanych przez niego czynności;
- śledzenie zdarzeń związanych z bezpieczeństwem, np. nieudane próby logowania, próby sięgania do nieprzydzielonych zasobów;
- niemożność ponownego wykorzystania pewnym obiektów, co zapobiega ich wykorzystaniu przez kogoś, kto podejrzwał wykonywaną na nich operację;
- szyfrowanie danych przesyłanych w sieci komputerowej.

Na szczególnie niebezpieczeństwo są narażone komputery podłączone do Internetu. Z tego względu pojawiło się specjalne rozwiązanie nazywane zaporą ogniową (*firewall*). Zapora ogniowa to odpowiednio oprogramowany komputer, ulokowany między lokalną siecią komputerową a podsiecią transmisji danych. Zadaniem komputera jest śledzenie i analiza danych wymienianych pomiędzy siecią lokalną a Internetem. Ściana ogniowa przegląda pocztę elektroniczną i wyszukuje w niej wirusy, sprawdza tożsamość i prawa dostępu zdalnych użytkowników, prowadzi dziennik, rejestrując zdarzenia związane z bezpieczeństwem, i alarmuje administratora w przypadku wykrycia ustalonego typu zdarzeń.

Banki są przykładem instytucji, które z oczywistych powodów mają najbardziej rozbudowane środki bezpieczeństwa. W tym przypadku, niezależnie od bezpieczeństwa systemów, chodzi również o niezawodne (bezwaryjne) i ciągłe (przez całą dobę we wszystkie dni roku) świadczenie usług. Mamy tu kilka wzajemnie się wspomagających kategorii środków zabezpieczających:

- fizyczne — urządzenia antywłamaniowe, sejfy, alarmy, urządzenia ochrony przeciwpożarowej;
- techniczne — urządzenia podtrzymujące zasilanie, karty magnetyczne i mikroprocesorowe, urządzenia do identyfikacji osób np. na podstawie linii papilarnych, głosu, siatkówki oka, urządzenia wykorzystywane do tworzenia kopii zapasowych, zapory ogniowe,

sprzętowe blokady dostępu do klawiatur, napędów dysków, dublowanie centrów obliczeniowych i baz danych;

- programowe — środki kontroli dostępu do systemu i do zasobów, dzienniki systemowe pozwalające na późniejszą identyfikację działalności użytkowników, programy monitorujące bieżącą pracę użytkowników systemu, mechanizmy identyfikacji wykonawców określonych operacji, programy antywirusowe, programy wykrywające słabe hasła istniejące w systemie;
- kryptograficzne;
- organizacyjne — opracowanie polityki bezpieczeństwa, stworzenie modelu i wdrażanie systemu zabezpieczeń, monitorowanie systemu.

Użytkownikom komputerów podłączonych do Internetu, poza przedstawionymi wyżej ogólnymi zasadami, warto przypomnieć o dwóch prostych zasadach, które należy stosować na co dzień. Po pierwsze, należy dokonywać systematycznie, najlepiej każdego dnia, aktualizacji eksploatowanego systemu operacyjnego. Mając legalne oprogramowanie, można tego dokonywać zdalnie za pomocą Internetu; producenci systemów operacyjnych, np. *Microsoft*, świadczą takie usługi. To właśnie wykryte niedoskonałości systemów operacyjnych są podstawą destrukcyjnego działania hakerów. Po drugie, należy w podobny sposób dokonywać aktualizacji zainstalowanego na komputerze programu antywirusowego.

Kryptografia

Szyfrowanie danych jest podstawowym środkiem ochrony *poufności* danych przesyłanych w Internecie. Kryptografia jest dziedziną wiedzy, która dostarcza środków do *szyfrowania* i *deszyfrowania* danych.

Dane dowolnej postaci: tekstowe, obrazowe czy dźwiękowe są reprezentowane w pamięci komputera jako ciągi zero-jedynkowe. Przyporządkowanie danym pewnego ciągu zero-jedynkowego nazywamy *kodowaniem*. Np. pojedynczemu znakowi tekstowemu odpowiada, w zależności od wybranego sposobu kodowania, pewien unikalny ciąg 7- lub 8-bitowy. Taki ciąg nazywa się kodem danego znaku. Mając pewien ciąg bitów i wiedząc, że zastosowaliśmy określony sposób kodowania, w jednoznaczny sposób przyporządkujemy temu ciągowi określony znak. Wyznaczanie znaku na podstawie znajomości jego kodu nazywamy *dekodowaniem*. Kodowanie i dekodowanie są czynnościami wzajemnie odwracalnymi.

Szyfrowanie odróżnia się od kodowania tylko tym, że sposób kodowania nie jest znany publicznie, ale jest znany tylko zainteresowanym użytkownikom. Dlatego ktoś, kto wejdzie w posiadanie danych, zakodowanych w nieznanym mu sposób, będzie miał trudności lub nawet nie będzie miał możliwości ich zdekodowania, czyli zdeszyfrowania.

Z utajnianiem przekazywanych wiadomości mamy do czynienia od najdawniejszych czasów. Jeden ze znanych sposobów szyfrowania pochodzi ze starożytności i jest znany jako szyfr Cezara. Warto ten prosty, można powiedzieć prymitywny, szyfr przedstawić tylko po to, aby wyjaśnić dalej przydatne pojęcia. Szyfrowanie dotyczy wyrazów tekstu i polega na tym, że pojedyncze litery alfabetu zastępuje się innymi literami tego samego alfabetu. Literę zastępuje się inną literą, która jest w porządku alfabetycznym przesunięta od danej litery w prawo o ustaloną liczbę pozycji.

Parametr szyfrowania **ps** oznacza przesunięcie. Wiersz oznaczony wartością parametru **ps** = 0 (brak przesunięcia) przedstawia uporządkowaną alfabetycznie listę 26 dużych liter alfabetu łacińskiego. Następne wiersze przedstawiają tę samą listę przesuniętą w lewo modulo 26 o 2 i o 3 pozycje. Modulo 26 oznacza, że pozycje na początku listy po przesunięciu wędrują na koniec. Te same kolumny, ponumerowane od 1 do 26, wyznaczają odpowiedniość pomiędzy literą w wyrazie oryginalnym a literą w wyrazie zaszyfrowanym.

Tab. 7.1

Tabela szyfrowania kodem Cezara

ps	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
0	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
2	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
3	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Parametr **ps** może przyjmować wartości od 1 do 25. W zależności od wyboru parametru przesunięcia otrzymamy różne kody zaszyfrowanych słów.

Tab. 7.2

Przykłady szyfrowania słów

Oryginalne słowo	Zaszyfrowane (ps = 2)	Zaszyfrowane (ps = 3)
KOD	MQF	NRG
SZYFR	UBAHT	XCBIU

Zdeszyfrowanie zakodowanego słowa, przy założeniu znajomości parametru szyfrowania **ps**, nie stanowi trudności. Postępowanie polega na utworzeniu tabeli deszyfrowania z wartością parametru deszyfrowania **pd** = 26 – **ps**. W przypadku gdybyśmy nie znali parametru szyfrowania, zdeszyfrowanie wyrazu nie będzie stanowiło problemu, gdyż wystarczy przebadać 25 różnych możliwych wartości parametru **pd**.

Szyfr Cezara można ogólnie scharakteryzować następująco. Jest on zdefiniowany przez ustalony algorytm przetwarzania tekstów. Użycie algorytmu wymaga ustalenia parametru przesunięcia. Algorytm ten jest jednakowy dla szyfrowania i deszyfrowania. Jeżeli przy szyfrowaniu używa się parametru szyfrowania **ps**, to przy deszyfrowaniu — parametru **pd**.

Używając współczesnej terminologii, możemy powiedzieć, że parametr **ps** jest *kluczem szyfrującym* używanym przez nadawcę, a parametr **pd** — *kluczem deszyfrującym* używanym przez odbiorcę wiadomości. Zauważmy też, że gdyby do zaszyfrowania wiadomości użyć klucza **pd**, to do odszyfrowania tej wiadomości byłby potrzebny klucz **ps**. Zatem **ps** i **pd** stanowią parę kluczy, których w dowolnej kolejności można użyć do szyfrowania i deszyfrowania.

Współcześnie stosowane systemy szyfrowania są oczywiście bez porównania bardziej złożone, ale tak samo jak system szyfrowania Cezara są zdefiniowane przez pewien algorytm oraz klucze szyfrowania i deszyfrowania. Algorytmy są w większości przypadków powszechnie znane, natomiast w tajemnicy utrzymuje się używane klucze. Specyfika systemów szyfrowania algorytmów polega na tym, że pomimo znajomości algorytmów, ale nieznaności kluczy, zdeszyfrowanie tekstu, nawet przy pomocy bardzo wydajnych komputerów, wymaga bardzo długiego czasu.

Współczesne systemy szyfrowania, opierając się na teorii arytmetyki liczb, wykorzystują szczególne własności liczb pierwszych, dlatego klucze również są dużymi liczbami pierwszymi. Duże liczby pierwsze to takie, których przedstawienie w notacji dziesiętnej wymaga przynajmniej kilkuset pozycji. Znajdowanie takich liczb, wbrew pozorom, jest zadaniem bardzo złożonym obliczeniowo, co powoduje, że znalezione liczby pierwsze są pilnie strzeżone w sejfach różnych agencji rządowych.

Istnieje wiele systemów szyfrowania, wśród nich wyróżnia się dwie grupy: szyfrowanie *symetryczne* i *asymetryczne*. Szyfrowanie symetryczne opiera się na pojedynczym kluczu, wspólnym dla nadawcy i odbiorcy. Szyfrowanie asymetryczne opiera się na dwóch kluczach: oddzielnym dla nadawcy i dla odbiorcy.

Szyfrowanie symetryczne jest mniej bezpieczne, gdyż wymaga przekazania klucza od nadawcy do odbiorcy. Odbywa się zwykle poza siecią, co jest czynnością narażoną na niebezpieczeństwo jego podejrzenia. Natomiast szyfrowanie asymetryczne nie wymaga przekazywania klucza nadawcy do odbiorcy. Przykładem systemu szyfrowania opartego o szyfrowanie symetryczne jest standard DES, a asymetrycznego — standard RSA.

System DES (*Data Encryption Standard*), opracowany w końcu lat siedemdziesiątych, jest jednym z najpopularniejszych systemów symetrycznego szyfrowania danych, służącym m.in. do kodowania załączników w poczcie elektronicznej.

Algorytm opiera się na 56-bitowym tajnym kluczu, który wykorzystywany jest do szyfrowania 64-bitowych bloków danych. Operacja przebiega w kilku, kilkunastu etapach, podczas których tekst wiadomości ulega wielokrotnym przeobrażeniom. Tak jak w każdej metodzie kryptograficznej posługującej się pojedynczym kluczem, musi być on znany zarówno nadawcy, jak i odbiorcy.

Ponieważ dla każdej wiadomości klucz wybierany jest losowo spośród ponad 10^{15} możliwych kluczy, wiadomości szyfrowane przy pomocy algorytmu DES do niedawna uchodziły za niemożliwe do złamania. Pod koniec lat dziewięćdziesiątych przeprowadzono skuteczne eksperymenty złamania kodu, pokazano, że za pomocą specjalnie skonstruowanego komputera każdą wiadomość daje się rozszyfrować w ciągu kilku dni. Nie wyklucza to jednak dalszego zastosowania systemu, ale oczekuje się przejścia na klucze o większej długości, przynajmniej o długości 128 bitów.

System RSA (skrót od nazwisk autorów: Rivest, Shamir, Adleman), również opracowany w końcu lat siedemdziesiątych, jest przykładem szeroko rozpowszechnionego systemu szyfrowania asymetrycznego. Algorytm RSA generuje unikalne pary kluczy: klucz publiczny i zależny od niego klucz prywatny. Pierwszy z nich służy do szyfrowania wiadomości kierowanych do właściciela kluczy i, co za tym idzie, jest powszechnie dostępny. Klucz prywatny jest tajny i tylko przy jego pomocy można odszyfrować to, co zostało zaszyfrowane kluczem publicznym.

Podstawową zaletą RSA jest odporność na złamanie szyfru przy dzisiejszym stanie wiedzy i technologii, wysoki poziom bezpieczeństwa zapewniają klucze o długości 4096 bitów, i takie też najczęściej stosowane są w praktyce (dla porównania: 56 bitom algorytmu DES odpowiada 512 bitów dla RSA). Wadą algorytmu RSA jest jego złożoność obliczeniowa.

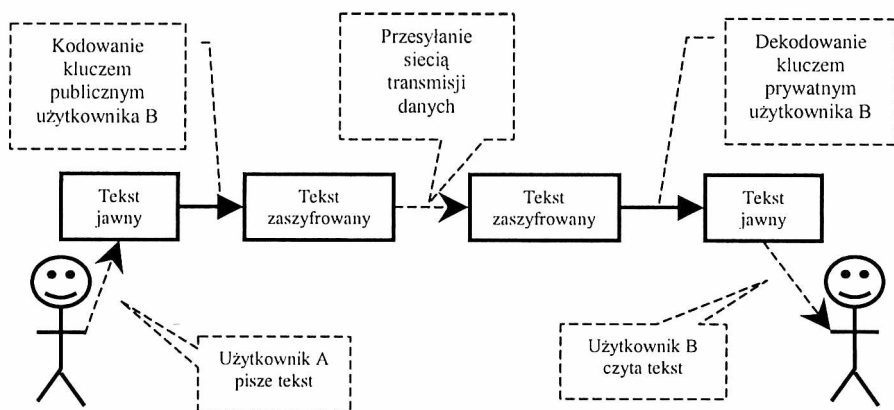
ok. tysiąc razy większa od algorytmu DES. RSA jest wkomponowany m.in. w przeglądarki *Netscape Navigator* oraz *Internet Explorer*.

Łącząc zalety obu podejść, stworzono hybrydowe systemy szyfrowania. Jednym z nich jest, opracowany na początku lat dziewięćdziesiątych, system PGP (*Pretty Good Privacy*, czyli *całkiem niezła prywatność*). PGP dostępny jest dla wielu systemów operacyjnych, m.in. *Windows* oraz różnych odmian systemu *Unix*, a także programów poczty elektronicznej, np. *Microsoft Outlook*, *Eudora*.

Poniższy schemat pokazuje proces kodowania, przesyłania i dekodowania wiadomości w oparciu o system szyfrowania asymetrycznego. Odpowiedni schemat dla szyfrowania symetrycznego różniłby się tylko tym, że oba klucze — publiczny i prywatny — byłyby takie same.

Rys. 7.1

Schemat przesyłania danych oparty o szyfrowanie asymetryczne



Omówione schematy szyfrowania gwarantują wysokie prawdopodobieństwo poufności przesyłanych danych. Nie gwarantują natomiast *integralności*, czyli nie zapobiegają modyfikacji danych podczas przesyłania, ale umożliwiają jej wykrycie. W tym celu przesyłaną wiadomość rozszerza się w taki sposób, że do jawnego tekstu dołącza się dodatkową daną kontrolną, nazywaną też skrótem wiadomości. Skrót wylicza się na podstawie tekstu jawnego, stosując powszechnie znane, jawne algorytmy. Odbiorca rozszerzonej wiadomości po jej odszyfrowaniu wydziela z niej oba fragmenty: tekst zasadniczy i skróty. Następnie wylicza skróty otrzyma-

nego tekstu i porównuje go ze skrótem w odebranej wiadomości. Jeżeli oba skróty są różne, to oznacza, że odebrana wiadomość uległa modyfikacji podczas jej przesyłania siecią transmisji danych.

Podpis elektroniczny

Przesyłając dane, oczekujemy, że zostaną one właściwie odczytane tylko przez wskazanego przez nas odbiorcę, a nie przez postronnego użytkownika, nawet gdyby udało mu się je podejrzeć. Odbiorca chce mieć przekonanie, że otrzymane dane zostały wysłane od zidentyfikowanego nadawcy — chodzi o *uwierzytelnienie* nadawcy, a dodatkowo chciałby, aby nadawca nie mógł zaprzeczyć temu, że to właśnie on był ich nadawcą — chodzi o *niezaprzeczalność* danych.

Koncepcja szyfrowania asymetrycznego jest podstawą podpisu elektronicznego, nazywanego także podpisem cyfrowym. Przypomnijmy — w systemach szyfrowania asymetrycznego każdy użytkownik sieci może dysponować dwoma kluczami szyfrowania: jeden, prywatny, pozostaje znany tylko jemu, a drugi, publiczny, jest znany wszystkim użytkownikom sieci. Klucze te stanowią parę, która jest używana w procesie szyfrowania i deszyfrowania tych samych danych. Każdy, kto chce do danego użytkownika przesłać poufną, przeznaczoną tylko dla niego informację, szyfruje ją, używając do tego klucza publicznego odbiorcy. Zaszyfrowaną informację może odszyfrować tylko posiadacz klucza prywatnego, czyli odbiorca.

Taki sposób szyfrowania zapewnia poufność przesyłania informacji, natomiast nie gwarantuje tego, że odbierający ma pewność, że informacja została przekazana od tego użytkownika, który określa się jako nadawca. Zauważmy, że gdyby nadawca informacji zamiast szyfrowania kluczem publicznym odbiorcy zaszyfrował ją swoim kluczem prywatnym, to wówczas odbiorca mógłby ją odszyfrować kluczem publicznym nadawcy. Ta właśnie obserwacja legła u podstaw podpisu elektronicznego, gdyż odbiorca tak zaszyfrowanej informacji może mieć wiarygodne przekonanie o tożsamości nadawcy.

Schemat przesyłania poufnej informacji wraz z podpisem elektronicznym przedstawia się następująco:

- nadawca przygotowuje wiadomość i szyfruje ją dwukrotnie — najpierw kluczem publicznym odbiorcy, a następnie swoim kluczem prywatnym;

- zaszyfrowana wiadomość otrzymana przez odbiorcę jest deszyfrowana — najpierw kluczem publicznym nadawcy, a następnie kluczem prywatnym odbiorcy.

Zapewnia to obu komunikującym się partnerom poufność wymienianej wiadomości, a odbiorcy zapewnia nie tylko uwierzytelnienie nadawcy, ale też nie pozwala nadawcy na zaprzeczenie, że to właśnie on był źródłem odebranej wiadomości.

Stosowanie podpisu elektronicznego ma praktyczny wymiar. Zakres i prawne konsekwencje jego stosowania w Polsce są określone ustawą o podpisie elektronicznym z dnia 18 września 2001 r. (Dz. U. z 2001 r. Nr 130, poz. 1450). Ustawa zrównuje pod względem prawnym podpis elektroniczny z podpisem własnoręcznym i ustala zasady funkcjonowania centrów certyfikacyjnych — instytucji zaufania społecznego, które świadczą usługi przydziału kluczy oraz niezbędnego oprogramowania umożliwiającego ich stosowanie użytkownikom Internetu.

Elementy netykiety

Zachowania użytkowników Internetu, stanowiących kilkusetmilionową, ponadnarodową grupę społeczną, wymagają przestrzegania — jak w każdej innej grupie społecznej — elementarnych zasad etycznych. Oczywiście, Internet nie wprowadza nowych kategorii i pojęć etycznych, raczej wymaga przystosowania do potrzeb swoich użytkowników powszechnie znanych zasad odpowiedzialności i zdrowego rozsądku. Sprowadza się to do pewnych zaleceń nazywanych *netykietą* (*net* — sieć i *etykieta*). Podane niżej zasady służą tylko uświadomieniu konsekwencji pewnych zachowań użytkowników i nie są normą prawną.

Internet jest federacją sieci. W każdej z nich mogą obowiązywać lokalne reguły i zwyczaje, których użytkownik powinien przestrzegać.

Każdy użytkownik sieci powinien być świadomy odpowiedzialności za swoje działania przy dostępie do rozległych zasobów Internetu: usług, komputerów, systemów i ludzi. Użytkownik pracujący w Internecie wnosi obciążenie do swojej sieci lokalnej i do sieci współpracujących, posiadając prawa dostępu do określonych zasobów. Należy zatem z tego prawa korzystać w sposób nie utrudniający pracy innym użytkownikom.

Przestrzeganie pewnych zasad to nie tylko troska o innych użytkowników, ale również dbałość o własne zasoby.

Przykładem są zalecenia dotyczące obsługi poczty elektronicznej w przypadku, gdy skrzynka pocztowa jest ulokowana w zdalnym serwerze, a nie w komputerze osobistym użytkownika.

- pocztę należy przeglądać codziennie i nie przekraczać wyznaczonego limitu pamięci; brak wolnej pamięci w skrzynce może uniemożliwić odbiór kolejnego listu;
- przejrane listy należy usuwać lub przesyłać do pamięci własnego komputera;
- nie można zakładać, że do skrzynki pocztowej ma wyłącznie dostęp dany użytkownik; mogą mieć do nich dostęp również osoby posiadające wyższe uprawnienia systemowe, dlatego nie należy przechowywać w niej szczególnie poufnych listów;
- niezależnie od tego, gdzie jest umieszczona skrzynka pocztowa, należy rutynowo sprawdzać, czy przychodzące listy nie zawierają wirusów.

Prowadząc korespondencję elektroniczną, zwłaszcza w ramach grup dyskusyjnych, należy:

- wypowiedzi formułować zwięźle i na temat;
- zachować ostrożność w przypadku stwierdzeń kontrowersyjnych, pamiętając, że mogą one zostać upublicznione;
- podawać źródła i referencje;
- ograniczać uczestnictwo w grupach dyskusyjnych do niezbędnego minimum, tak aby uniknąć kłopotu przeglądania dziesiątek czy setek codziennie nadchodzących listów i ograniczyć zbędny ruch w sieci;
- wypisywać się lub zawieszać subskrypcję we wszystkich listach dyskusyjnych na okres dłuższej, niekiedy nawet kilkudniowej, nieobecności;
- w stopce przesyłki pocztowej załączać swoje dane: imię, nazwisko, stanowisko, komórkę organizacyjną, adres e-mail, dodatkowo można podać adres pocztowy i numer telefonu lub faksu.

Szczególnego podkreślenia wymaga przestrzeganie warunków licencji i praw autorskich (*copyright*) używanego oprogramowania. Dotyczy to zarówno oprogramowania kupowanego poza siecią, jak i tego kopiowanego z sieci za pomocą usługi WWW lub FTP. Należy pamiętać, że naruszenie tej zasady jest nie tylko złamaniem etykiety, ale również złamaniem prawa (ustawa o prawie autorskim i prawach pokrewnych, Dz. U. z dnia 23 lutego 1994 r.).

Elementy ergonomii stanowiska pracy komputerowej

Komputer jest elementem stanowiska pracy. Korzystanie z niego wymaga odpowiedniego przygotowania stanowiska pracy i jego otoczenia. Kwestia ta jest szczególnie istotna w przypadku organizacji pracy w biurze czy urzędzie, gdzie użytkownik spędza przy komputerze większość czasu. Jest ona również przedmiotem *ergonomii*, czyli dziedziny wiedzy, która zajmuje się zasadami i metodami dostosowania warunków pracy do psychofizycznych właściwości człowieka. Poniżej przedstawimy główne zalecenia ergonomiczne dotyczące komputerowego stanowiska pracy.

Ze stanowiskiem pracy wiążemy przede wszystkim rozmieszczenie elementów systemu komputerowego, natomiast z jego otoczeniem — warunki panujące w pracowni komputerowej, takie jak oświetlenie czy wilgotność powietrza.

Podstawą aranżacji stanowiska jest stół. Jego parametry muszą być dobrane do rozmiarów pomieszczenia, w którym będzie stał, powinien też mieć ruchomy blat pod klawiaturę, pożądana jest regulacja wysokości blatu, ale zamiast tego wystarczy krzesło o regulowanej wysokości. Blat powinien być na tyle duży, by zapewnić wolne miejsce na materiały pisane i podstawowe urządzenia peryferyjne: monitor, klawiaturę, myszkę.

Monitor może być umieszczony na blacie stolika, albo lepiej na przystawianej do blatu, ruchomej podstawie pozwalającej na zmiany położenia monitora w trzech wymiarach. Monitor powinien być umieszczony tak, by na jego ekran bezpośrednio nie padało światło naturalne lub sztuczne i by pozwalał na patrzenie na dalszą część pokoju, dając chwilowy odpoczynek oczom. Istotny wpływ na zmęczenie użytkownika ma wielkość ekranu i jakość wyświetlanego obrazu. Należy pamiętać, że monitor, zwłaszcza kineskopowy, jest źródłem szkodliwego promieniowania elektromagnetycznego, a także pola magnetycznego i elektrostatycznego. Stopień eliminacji tych czynników zależy głównie od jakości samego monitora, od wbudowanych filtrów. Niezależnie od tego zaleca się, aby odległość od ekranu do oczu użytkownika wynosiła przynajmniej 50 cm. Innym ważnym zaleceniem jest to, aby bezpośrednio z tyłu monitora kineskopowego nie urządzać innego stanowiska pracy, gdyż wyrzutnia lampy kineskopowej jest również źródłem szkodliwego promieniowania.

Szczegółowe wymogi dotyczące bezpieczeństwa i higieny pracy oraz ergonomii stanowisk pracy wyposażonych w monitory ekranowe zawiera

Rozporządzenie Ministra Pracy i Polityki Socjalnej z 1 grudnia 1998 r. (Dz. U. z dnia 10 grudnia 1998 r.).

Ułożenie klawiatury powinno w jak najmniejszym stopniu obciążać dłonie i zmuszone do opierania się nadgarstki. Spotykamy obecnie dwa rodzaje klawiatur — tradycyjne, z prostym układem klawiszy, i ergonomiczne, o różnych „łamanych” kształtach. Klawiatury ergonomiczne są lepsze ze względu na naturalny układ dłoni, co, jak wynika z badań, przekłada się na mniejszą liczbę popełnianych błędów przy pisaniu tekstów. W przypadku gdy z klawiatury wprowadzamy duże ilości tekstu, niezbędna jest, przymocowywana do blatu, podstawka utrzymująca przepisywany tekst w pozycji pionowej.

Podczas pracy siedzimy na krześle, które powinno mieć regulację wysokości siedziska, kąta nachylenia oparcia i wysokości poręczy. Jeżeli nie dysponujemy stołem z regulowaną wysokością blatu, dodatkiem do krzesła powinien być podnózek, pozwalający na wygodne ułożenie stóp.

Aranżacja pracowni komputerowej w zwykłych pokojach biurowych powinna zapewniać powierzchnię przynajmniej 6 m² na jeden komputer. Ściany i wykładziny powinny mieć barwy ochronne dla wzroku, wykładziny podłogowe powinny być niepodatne na elektryzację. Niezbędna jest też wentylacja, a jeszcze lepiej klimatyzacja, zapewniająca temperaturę 21-23°C i wilgotność względną 60-80%. Oświetlenie pomieszczenia, naturalne lub sztuczne, powinno umożliwiać odczytywanie znaków na ekranie bez wymuszania akomodacji wzroku. Wreszcie, w pomieszczeniu monitorów nie powinno być źródeł wibracji ani hałasów. Ostatnia część wymagań odnosząca się do pracowni komputerowej w zasadzie nie odbiega od wymagań stawianych dobrze zorganizowanej pracowni biurowej.

Inne zalecenia dotyczące komputerowych stanowisk pracy można znaleźć w materiałach Centralnego Instytutu Ochrony Pracy (<http://www.ciop.waw.pl/>).

Na rynku międzynarodowym stosuje się normy różnych organizacji, które w różnym zakresie odnoszą się do komputerowych stanowisk pracy. Przykładem norm dotyczących monitorów, wobec których wymagania są najostrejsze, są serie norm TCO (*The Swedish Confederation of Professional Employees*) oraz Nemko GS (*Geprüfte Sicherheit*). Uzyskanie certyfikatu TCO'99 świadczy o najwyższej jakości monitora, natomiast certyfikatu Nemko GS — jest warunkiem dostępu na rynek niemiecki. W Polsce w najbliższym okresie, w związku z wejściem do Unii Europejskiej, zaczną obowiązywać normy europejskie.

VIII. BAZY DANYCH

Wprowadzenie

Jednym z celów posługiwania się komputerem jest gromadzenie i przechowywanie danych. Omawiany w rozdziale IV system plików stanowi przykład pewnego nagromadzenia danych, a dokładniej — nagromadzenia plików. System plików można byłoby traktować jako pewien wyspecjalizowany przykład bazy danych. Specjalizacja ta polega na tym, że system plików zawiera informacje w zasadzie tylko o nazwach plików. Mówiąc o bazach danych, mamy na myśli nie pliki, a raczej ich zawartość i to, w jaki sposób zawartość ta odnosi się do danej dziedziny zastosowań.

Przykładem prostych baz danych są spisy towarów, rozkłady jazdy, ewidencje pracowników, wykazy płatnicze, wszelkiego rodzaju rejestry bankowe, kartoteki urzędowe, biblioteki, katalogi, książki telefoniczne, szpitalne historie chorób, dzienniki lekcyjne, wykazy osób podejrzanych, księgi parafialne. Wszystkie one muszą spełniać jeden warunek: ich tworzenie, przechowywanie, uaktualnianie i przeszukiwanie powinno się odbywać w komputerze.

Dla ilustracji bazy danych rozpatrzmy przykład. Załóżmy, że interesuje nas baza danych gromadząca dane adresowe, sporządzana np. na potrzeby domowe. Jej zawartością mają być dane o różnych osobach, a w ramach tych danych mogą być uwzględnione następujące atrybuty:

- imię,
- nazwisko,
- adres domowy,
- telefon domowy,
- telefon służbowy,
- e-mail.

Dla tak określonej bazy danych łatwo można wyobrazić sobie jej wersję w postaci tabeli, zapisaną w notatniku.

Tab. 8.1
Przykład prostej bazy danych

Imię	Nazwisko	Adres domowy	Telefon domowy	Telefon służbowy	E-mail
Adam	Dacko	Świdnicka 13, Wrocław	2323244	3435632	
Barbara	Emska	Radom, Uciecha 21/3	4564661		b.emska@republika.com.pl
Celina	Falska	Zabrze, Elementarzowa		5321236	
....					

Poza nagłówkiem tabeli poszczególne wiersze odnoszą się do pojedynczych osób, liczba tych wierszy jest równa liczbie osób, o których zgromadzono dane. Kolumny tabeli odnoszą się do wybranych wcześniej atrybutów. Pola tabeli zawierają wartości danego atrybutu dla danej osoby.

Zapisując tabelę w notatniku, rozmiary pól dobieralibyśmy stosownie do wartości, jaką chcielibyśmy w tym polu umieścić. Podobnie w przypadku bazy danych zapisywanych w komputerze musimy określić charakter zawartości pola, wyznaczając typ i format. Typ określa dopuszczalny zbiór wartości, a format — formę zapisu wartości. Przykłady typów spotykanych w wielu systemach baz danych są następujące:

- typ znakowy (tekstowy) — jest ciągiem znaków z określonego alfabetu; format ogranicza się do podania maksymalnej długości takiego ciągu;
- typy liczbowe: całkowitoliczbowe i rzeczywiste — w przypadku liczb całkowitych format wynika z ograniczenia zakresu, a w przypadku liczb rzeczywistych może określać dodatkowo liczbę miejsc po przecinku;
- typ logiczny — jego wartościami są *tak* i *nie*;
- typ wyrażający datę lub datę i godzinę; formaty dotyczące daty mogą mieć różne postaci: 2003-07-24, 24 lipca 2003, 24 lip 03 itp.;
- typ walutowy — format dotyczy tu wyboru jednostki: zł, euro itp.

Warto zwrócić uwagę na to, że kolejność zapisu wierszy i kolumn nie zmienia informacji, jaką przedstawia baza danych. Kolejność ta ma jedynie wpływ na wygodę posługiwania się bazą. Np. przy wyszukiwaniu informacji o danej osobie wygodne byłoby uporządkowanie wierszy w kolejności alfabetycznej nazwisk, ale przy wyszukiwaniu takich danych na podstawie znajomości numeru telefonu wygodniejsze byłoby inne uporządkowanie wierszy.

Widać, że nie wszystkie pola tabeli są wypełnione. Może to oznaczać jedną z dwóch możliwości:

- dana osoba nie ma telefonu lub skrzynki poczty elektronicznej,
- nie znamy tego telefonu lub skrzynki poczty elektronicznej, mimo że dana osoba może je posiadać.

Definiując typ dla wartości danego atrybutu, powinniśmy jawnie określić, czy dopuszczamy niewypełnione pola w tabeli. Formalnie wyraża się tę możliwość przez wpisywanie w niewypełnione pole umownej wartości *NULL*.

Zwrócenia uwagi wymaga definicja wartości pola adresowego. Jego wartością jest tekst, w którym nie ma wydzielonych elementów składowych powszechnie rozumianego adresu, takich jak: miejscowość, kod pocztowy, ulica, numer domu, numer mieszkania. Pozostawiając możliwość dowolnego sposobu wypełnienia tego pola, tracimy nawet szansę sensownego uporządkowania wierszy względem zawartości tego pola, np. uporządkowanie wierszy w kolejności alfabetycznej nie daje nawet gwarancji, że będzie to kolejność względem miejscowości zamieszkania. Ponadto, traktując wartość tego pola jako tekst, nie można bez analizy tego tekstu stwierdzić, czy adres jest kompletny.

Wskazane niedogodności biorą się stąd, że wartości pola są wartościami nieelementarnymi. Dlatego definiując wartości pól, należy zadbać o to, aby stanowiły one wartości elementarne, nierozkładalne na elementy składowe.

Tabela zawiera wypełnioną pewną liczbę wierszy, które reprezentują aktualny stan bazy danych. Jest on odzwierciedleniem naszej wiedzy o pewnym wycinku rzeczywistości. Oczywiście, oczekujemy, że stan bazy danych jest zgodny z rzeczywistością, gdyż inaczej sens jej tworzenia i wykorzystywania byłby wątpliwy.

Stan ten może się zmieniać wskutek aktualizacji tabeli, przy:

- dopisywaniu nowych wierszy, czyli wprowadzaniu danych o nowych osobach;
- kasowaniu aktualnie zapisanych wierszy, czyli wykreślaniu pewnych osób z bazy;
- aktualizowaniu poszczególnych pól pamiętanych wierszy, czyli aktualizowaniu danych o pamiętanych osobach.

W ramach danego stanu możemy z tabeli odczytywać te dane, które nas interesują, np. adres osoby o określonym nazwisku lub nazwisko osoby na podstawie numeru telefonu. Może być nam także przydatne sporządzenie pewnego wyciągu z tabeli, np. tylko nazwisk osób i ich telefonów domowych.

Wymienione wyżej operacje na tabeli określa się mianem operacji bazodanowych. Łatwo zgodzić się ze stwierdzeniem, że nie powinny być jednocześnie wykonywane dwie różne operacje, które dokonują zmiany stanu bazy danych. Dopuszczenie takiej możliwości mogłoby doprowadzić do nowego stanu niespójnego z rzeczywistością, np. powstania wiersza o niepoprawnych danych.

Systemy baz danych

Bazę danych można określić nieformalnie jako pewien uporządkowany zbiór trwałych danych — magazyn danych — wraz z mechanizmami dostępu i wyszukiwania, mający określone przeznaczenie.

Występujące w tym określeniu pojęcia mają następujące znaczenie:

- *uporządkowanie* danych — dane mają jawnie określoną strukturę, a sposób ich reprezentacji i pamiętania służy temu, by można efektywnie wyszukiwać potrzebne dane, modyfikować, usuwać lub dołączać nowe;
- *trwałość* danych — mogą być one przechowywane przez dowolnie określony czas;
- *przeznaczenie* bazy — gromadzone dane są związane z funkcjonowaniem jednoznacznie określonego wycinka rzeczywistości — firm, organizacji, instytucji.

Bazy danych służą rejestracji faktów i zdarzeń zachodzących w ustalonym wycinku rzeczywistości. Dzięki temu baza danych może być wykorzystana do analizy dotychczasowego funkcjonowania i pomocna przy podejmowaniu decyzji dotyczących przyszłego funkcjonowania wybranego wycinka rzeczywistości.

Dane przechowywane w bazie danych mogą mieć postać tekstu, liczb, a także obrazów, dźwięków i obrazów ruchomych. W przypadku gdy dane przechowywane w bazie mają różnorodny charakter, mówi się o *multimedialnych* bazach danych.

W zależności od tego, jaki charakter mają zgromadzone dane, wyróżnia się: *bazy tradycyjne* i *bazy wiedzy*, nazywane też bazami *dedukcyjnymi* lub *inteligentnymi*.

Bazy tradycyjne stanowią zgromadzenie faktów, natomiast bazy wiedzy gromadzą fakty oraz reguły wnioskowania, na podstawie których można wyprowadzać nowe fakty.

W celu rozróżnienia bazy tradycyjnej i bazy wiedzy rozpatrzmy przykład. Niech tradycyjna baza danych gromadzi dane osobowe o poszczególnych osobach, zapamiętując informacje takie jak w poprzednim przykładzie i dodatkowo informacje o rodzicach oraz o dzieciach. Baza ta jest zbiorem faktów, fakty są zestawami informacji o pojedynczych osobach.

Na podstawie tak określonej bazy danych można zbudować bazę wiedzy, która gromadzi wszystkie fakty — dane osobowe, dołączając pewne reguły wnioskowania w oparciu o te fakty. Np. mogą to być reguły, które pozwalają na określanie dla danej osoby wszystkich jej przodków albo wszystkich jej potomków. Łatwo przekonać się, że mając dane osobowe zawierające informacje o rodzicach i dzieciach, można na tej podstawie wywnioskować, które osoby są jej przodkami (dziadkami, pradiadkami), a które potomkami (wnukami, prawnukami). Jeżeli stosując reguły, określimy dla danej osoby zbiór jej przodków i potomków, to jest to nowy fakt, który nie ma bezpośredniej reprezentacji w tradycyjnej bazie danych.

Wśród tradycyjnych baz danych, ze względu na sposób strukturyzacji danych, wyróżnia się dwa główne rodzaje baz: *relacyjne* i *obiektowe*.

Bazy relacyjne mają ponad 30-letnią historię, są szeroko rozpowszechnione, stanowią obecnie ponad 95% funkcjonujących baz danych. Swoją pozycję zawdzięczają temu, że bardzo dobrze odpowiadają potrzebom gospodarczym oraz są bardzo dobrze rozwiniętymi narzędziami informatycznymi. Bazy obiektowe, stanowiące pewnego rodzaju uogólnienie baz relacyjnych, znajdują się w początkowym okresie rozwoju. Można jednak oczekiwać, że w najbliższych 10 latach staną się mocnym konkurentem baz relacyjnych. W dalszym ciągu będziemy rozważać tylko relacyjne bazy danych.

Często spotyka się *rozproszone bazy danych*. Pod tym określeniem rozumie się system bazy danych, w którym dane są rozmieszczone w komputerach znajdujących się w różnych miejscach. Rozmieszczenie elementów rozproszonej bazy danych może dotyczyć jednego biura, budynku, lecz także obszarów rozległych geograficznie, np. krajowy system oddziałów banku. Z punktu widzenia użytkownika rozproszona baza danych jest widziana tak jakby była umieszczona tylko w jednym komputerze.

Pojęcie bazy danych jest używane w dwóch znaczeniach. W znaczeniu szerokim baza danych to tyle, co system bazy danych, w znaczeniu wąskim — to tylko element systemu bazy danych.

System bazy danych składa się z dwóch elementów:

- uporządkowanego zestawu danych, nazywanego zwykle *bazą danych*,
- *systemu zarządzania bazą danych* (SZBD).

Bazę danych, jako uporządkowany zbiór danych, przedstawia się w postaci zbioru powiązanych ze sobą tabel. Tabele te i powiązania między nimi, zawsze odnosząc się do pewnego wycinka rzeczywistości — osób, instytucji, organizacji społecznych lub gospodarczych, powinny wiernie odzwierciedlać przedmiot zainteresowania użytkowników bazy.

Określenie struktury gromadzonych danych nie wystarcza do zagwarantowania ich zgodności z wycinkiem rzeczywistości, do której się odnoszą — dane muszą spełniać jeszcze dodatkowe warunki. Zgodność danych przechowywanych w bazie z wycinkiem rzeczywistości można definiować przez podanie *statycznych* i *dynamicznych warunków* (wiązków) *integralności*.

Statyczne warunki integralności określają te własności danych, które muszą być spełnione w dowolnym stanie bazy danych. Przykładem statycznego warunku integralności w bazie danych przechowujących dane osobowe może być wymóg, aby w przypadku gdy baza zawiera dane rodziców i dzieci, data urodzenia każdego z rodziców była wcześniejsza od daty urodzin ich dzieci. Jeżeli w takiej bazie jest data urodzenia i jednocześnie jest podawany wiek osoby, to — w danym momencie czasu — wiek osoby powinien być różnicą, wyrażoną w latach, pomiędzy datą bieżącą a datą urodzenia. Warunki te są odzwierciedleniem opisywanej rzeczywistości.

Dynamiczne warunki integralności określają dla danego stanu bazy danych dopuszczalne stany, do których może przejść baza danych w wyniku realizacji dowolnej transakcji; inaczej: warunki dynamiczne określają dopuszczalne przejścia pomiędzy kolejnymi stanami bazy danych. Przykładami dynamicznych warunków integralności, w tej samej bazie przechowującej dane osobowe, może być wymóg, aby każda modyfikacja stanu bazy danych polegająca na aktualizacji danych względem bieżącej daty powodowała, aby dla każdej osoby wiek był równy różnicy pomiędzy bieżącą datą a datą urodzenia. Jeżeli w przedsiębiorstwie obowiązuje zasada, że zmiana wynagrodzenia pracownika może polegać tylko na podwyżce, to w osobowej bazie danych przekłada się to na warunek, by każda aktualizacja pola dotyczącego zarobku danej osoby wprowadzała wartość większą od dotychczasowej wartości tego pola.

System zarządzania bazą danych można określić jako specjalistyczny system operacyjny, który:

- przydziela użytkownikom dostęp do bazy i określa ich prawa dostępu, np. prawo czytania danych lub prawo czytania i modyfikacji, prawo dostępu tylko do określonych fragmentów bazy (w zależności od statusu: zwykłego użytkownika lub administratora bazy danych); jest to czynione, podobnie jak w systemach operacyjnych, w celu zapewnienia bezpieczeństwa i poufności danych;
- udziela odpowiedzi na pytania użytkowników, wyszukując w bazie danych wskazanych przez nich informacji; pytania są formułowane przez użytkowników w pewnym języku formalnym, który umożliwia sformułowanie własności dotyczących wyszukiwanych danych;
- organizuje jednoczesną obsługę wielu użytkowników, zapewniając poufność i spójność bazy danych, inaczej: realizuje transakcje określone przez użytkowników bazy danych;
- zarządza bazą danych, co obejmuje tworzenie i aktualizację bazy danych, a także pieczę nad bazą w przypadku awarii systemu komputerowego, w którym funkcjonuje baza danych; w celu zabezpieczenia przed skutkami awarii SZBD zapamiętuje pewien stan bazy i stan realizacji transakcji, dzięki czemu po usunięciu awarii możliwe jest odtworzenie tego stanu i kontynuacja realizacji transakcji.

Dane zgromadzone w bazie danych są dostępne dla użytkownika, który może *wyszukać* interesujące dane, a także może je *zmodyfikować* (np. zmienić wartość danej prostej), *usunąć* lub *dodać* do bazy nowe dane — mówimy, że użytkownik może dokonywać pewnych *operacji bazodanowych*, zwanych *transakcjami*.

Transakcje mogą być wykonywane jednocześnie przez wielu użytkowników, ale w taki sposób, by wzajemnie ze sobą nie kolidowały. Np. nie można dopuszczać do tego, aby jednocześnie ta sama dana była przez jednego użytkownika odczytywana, a przez drugiego użytkownika modyfikowana. W związku z tym od transakcji wymaga się następujących czterech własności:

- *niepodzielności*, co oznacza, że transakcja może być wykonana w całości albo nie wykonana wcale (nie dopuszcza się przerywania wykonywania transakcji),
- *spójności*, czyli takiego zachowania, aby wykonanie transakcji nie naruszyło warunków integralności bazy danych,

- *izolacji*, co oznacza, że wynik realizacji danej transakcji jest niezależny od innych transakcji, które są realizowane w tym samym czasie,
- *trwałości*, co oznacza, że wynik wykonania transakcji jest trwale zachowany w bazie danych.

W danej chwili baza danych jest w pewnym *stanie*, to znaczy jest konkretnym, uporządkowanym zbiorem danych. Stan bazy danych może zmieniać się na skutek wykonywanych transakcji. Skasowanie danej, dołączenie nowej danej lub modyfikacja pamiętanej danej zmienia zbiór pamiętanych danych, a więc stan bazy danych, natomiast wyszukanie i odczytanie pamiętanych danych stanu tego nie zmienia.

Przykładowa baza danych

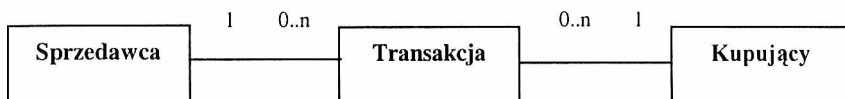
Rozważmy firmę handlową, w której pracuje pewna liczba sprzedawców. Przeprowadzają oni transakcje kupna-sprzedaży z klientami firmy. Założmy, że firma postanawia założyć bazę danych służącą do gromadzenia danych o sprzedawcach, klientach i przeprowadzonych transakcjach. Nasuwające się rozwiązanie polega na zdefiniowaniu oddzielnych tabel dla sprzedawców, klientów i transakcji, nazywanych dalej: *Sprzedawca*, *Kupujący*, *Transakcja*. Należy oczywiście określić, jakie informacje będziemy gromadzić w tych tabelach, oraz ustalić, na czym mają polegać związki między tabelami.

Zacznijmy od stwierdzenia, że w realizacji pojedynczej transakcji uczestniczy jeden sprzedawca i jeden kupujący. Z punktu widzenia sprzedawcy chodzi o to, by znać transakcje zrealizowane z jego udziałem. Sprzedawca może być zaangażowany w wiele transakcji, w szczególnym przypadku, a tak jest na początku jego pracy w firmie, nie jest zaangażowany w żadną transakcję. Podobnie przedstawia się to z punktu widzenia kupującego w danej firmie: może być on uczestnikiem dowolnej liczby transakcji.

Przedstawione ustalenia można wyrazić graficznie w postaci *diagramu klas* (rys. 8.1). Prostokąty na diagramie są klasami. Klasa reprezentuje zbiór potencjalnie możliwych bytów, które mają te same właściwości. Pojedynczy byt należący do danej klasy nazywa się instancją klasy. Zatem instancjami klasy *Sprzedawca* będą konkretni sprzedawcy zatrudnieni w firmie, instancjami klasy *Kupujący* — konkretni klienci firmy, a klasy *Transakcja* — konkretne akty kupna-sprzedaży.

Linie łączące poszczególne klasy nazywa się asocjacjami. Wyrażają one związki pomiędzy instancjami odpowiednich klas. Asocjacja łącząca klasę *Sprzedawca* z klasą *Transakcja* ma znaczenie następujące. Z punktu widzenia instancji klasy *Sprzedawca*, czyli dowolnego konkretnego sprzedawcy, asocjacja wyraża, że może on uczestniczyć w dowolnej liczbie konkretnych transakcji — liczba transakcji wynosi od 0 do n , gdzie n jest dowolną całkowitą liczbą dodatnią. Ta sama asocjacja, z punktu widzenia instancji klasy *Transakcja*, czyli konkretnego aktu kupna-sprzedaży, oznacza, że po stronie klasy *Sprzedawca* musi uczestniczyć w niej dokładnie jeden sprzedawca. Podobnie można wyjaśnić znaczenie drugiej asocjacji, zachodzącej pomiędzy klasami *Transakcja* i *Kupujący*.

Rys. 8.1
Przykładowy diagram klas



W tworzonej bazie danych klasy będą reprezentowane odpowiednimi tabelami. Asocjacje pomiędzy klasami będą odzwierciedlone związkami pomiędzy tabelami, co będzie wymagać odpowiedniej konstrukcji tabel.

Przyjmijmy założenie, że każda tabela będzie zawierać jeden wyróżniony atrybut, nazywany *kluczem głównym* tabeli. Rola klucza głównego polega na tym, że jego wartość wskazuje jednoznacznie dokładnie jeden wiersz tabeli, inaczej: wartość klucza głównego jest unikalnym identyfikatorem wiersza. Klucze główne w poszczególnych tabelach będą nazywane: *IdSprzedawca*, *IdTransakcja*, *IdKupujący*, a wartościami kluczy będą liczby całkowite. Schematy tabel przedstawia się w postaci nazwy tabeli i listy jej atrybutów oraz ich typów:

Sprzedawca (*IdSprzedawca*: Całkowita,

Nazwisko: Tekst,

Data zatrudnienia: Data,

Pensja: Całkowita,

IdTransakcja: Całkowita)

Transakcja (*IdTransakcja*: Całkowita,

Towar: Tekst,

Data sprzedaży: Data,

IdSprzedawca: Całkowita

IdKupujący: Calkowita)

Kupujący (IdKupujący: Calkowita,

Nazwisko: Tekst,

Adres: Tekst,

IdTransakcja: Calkowita)

Para *IdSprzedawca: Calkowita* oznacza nazwę atrybutu *IdSprzedawca* i jego typ *Calkowita*. Podobne znaczenie mają pozostałe pary tej postaci.

W schemacie tabeli *Sprzedawca* obok atrybutu *IdSprzedawca*, który pełni rolę klucza głównego, znajduje się atrybut *IdTransakcja*, który jest w tej tabeli kluczem obcym. Klucz obcy pełni rolę wskaźnika (referencji) do tabeli *Transakcja*, to znaczy wskazuje ten wiersz tabeli *Transakcja*, gdzie znajdują się szczegóły tej transakcji, w której uczestniczył dany sprzedający. Podobna jest rola kluczy *IdSprzedawca* oraz *IdKupujący* w tabeli *Transakcja* i klucza *IdTransakcja* w tabeli *Kupujący*.

Przedstawiony niżej konkretny zestaw tabel jest ilustracją pewnego stanu bazy opartej na omówionych schematach.

Tab. 8.2
Sprzedawca

IdSprzedawca	Nazwisko	Data zatrudnienia	Pensja	IdTransakcja
1	Matul	1998	2700	5
2	Berkowski	2002	2100	4
3	Matul	1998	2700	3
4	Gondalski	2003	1550	2
5	Gondalski	2003	1550	1

Tab. 8.3
Transakcja

IdTransakcja	Towar	Data sprzedaży	IdSprzedawca	IdKupujący
1	Mondeo	2002-10-10	5	4
2	Brava	2002-03-21	4	5
3	Mondeo	2001-21-30	3	2
4	Mondeo	2002-12-09	2	2
5	Opel	2000-04-09	1	1

Tab. 8.4
Kupujący

IdKupujący	Nazwisko	Adres	IdTransakcja
1	Maresz	Wrocław	5
2	Janowski	Świdnica	3
3	Janowski	Świdnica	4
4	Świdoń	Wałbrzych	1
5	Łacki	Legnica	2

Przedstawiony przykład przedstawia przejście od modelu konceptualnego bazy danych, wyrażonego w postaci diagramu klas, do modelu logicznego, wyrażonego w postaci zbioru powiązanych tabel. Przejście to jest fragmentem projektowania baz danych, które obejmuje trzy fazy:

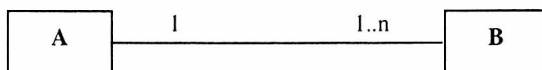
- projektowanie konceptualne,
- projektowanie logiczne,
- projektowanie fizyczne.

Z punktu widzenia nieinformatyka najważniejsza jest faza pierwsza, gdyż podjęte tu ustalenia decydują o sensowności dalszych faz, ale przede wszystkim decyduje o przydatności zastosowania projektowanej bazy. Mając ustanowiony model konceptualny, przejście do modelu logicznego, a później i fizycznego daje się zautomatyzować. Dla ilustracji tego stwierdzenia poniżej przedstawiono uproszczone zasady transformacji modelu konceptualnego w model logiczny:

1. Dla każdej klasy z diagramu klas tworzy się tabelę o takiej samej nazwie.

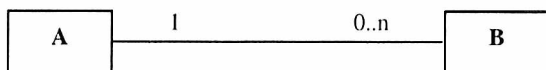
2. Atrybuty klasy stają się kolumnami tabeli. Zakłada się, że wartości atrybutów są wartościami elementarnymi. Wprowadza się kolumnę stanowiącą klucz główny.

3. Dla każdej asocjacji postaci:



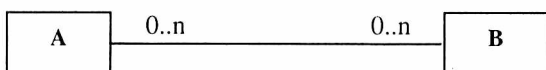
klucz główny dla klasy *B* wstawiamy, jako klucz obcy, nową kolumnę do tabeli reprezentującej klasę *A*.

4. Dla każdej asocjacji postaci:



tworzy się nowy schemat relacji, w którym umieszcza się klucze główne obu klas. Należy zwrócić uwagę, że — w odróżnieniu od reguły 3 — w tabeli *A* w kolumnie reprezentującej klucz obcy (klucz główny tabeli *B*) należy dopuścić wartość *NULL*. Wartość ta będzie oznaczać, że obiekt reprezentowany przez dany wiersz tabeli nie jest powiązany z żadnym obiektem reprezentowanym przez wiersze tabeli *B*.

5. Dla każdej asocjacji postaci:



tworzy się nową tabelę z ewentualnie ze sztucznym kluczem, która zawiera, jako klucze obce, klucze główne klas *A* oraz *B*.

Podane reguły dają na ogół dobre efekty, ale należy traktować je tylko jako zalecenia, gdyż w szczególnych przypadkach może okazać się, że korzystniejsze będą inne zasady transformacji.

Język manipulacji i zapytań

W celu korzystania z usług relacyjnej bazy danych użytkownicy komunikują się z systemem SZBD ustalonymi językami. Potrzebne są przynajmniej dwa języki: jeden służący do definiowania struktury bazy danych — DDL (*Data Definition Language*) i drugi do manipulowania na danych — DML (*Data Manipulation Language*).

Powstało wiele różnych języków definiowania i manipulacji na danych, wśród których szczególnie ważną rolę odegrały ISO (*International Standard Organization*) i ANSI (*American National Standard Institute*). Najszerszej rozpowszechnionym efektem prac jest standard SQL2 z 1993 r. Standard ten jest stosowany obecnie w niemal wszystkich systemach zarządzania bazami danych i określa m.in. wspólny język definiowania i manipulowania na danych. Język ten, choć obejmuje dwa języki, nazywa się SQL (*Structured Query Language*), czyli strukturalny język zapytań. Przykładowymi poleceniami, które można formułować w języku DDL, są:

- utworzenie bazy danych o określonej nazwie, np.:

```
CREATE DATABASE Pracownicy
```

- utworzenie tablicy o określonej nazwie, np.:

```
CREATE TABLE DaneOsobowe
```

```
(Nazwisko char(20) not NULL, Identyfikator int not NULL)
```

gdzie char(20) oznacza, że wartością atrybutu *Nazwisko* jest ciąg co najwyżej 20 znaków, not NULL — że wartość atrybutu musi być zdefiniowana, nie może być wartością pustą, int not NULL — że wartością atrybutu *Identyfikator* są liczby całkowite i wartości dla tego atrybutu muszą być zdefiniowane.

Do wcześniej zdefiniowanej tablicy można dodać nowy atrybut, np.:

```
ALTER TABLE DaneOsobowe
```

```
ADD Pensja int NULL
```

dołącza atrybut *Pensja*, który może być nieokreślony.

Utworzone tablice lub całą bazę danych można skasować, np.:

```
DROP TABLE DaneOsobowe
```

```
DROP DATABASE Pracownicy
```

Język DML służy do wstawiania, modyfikacji lub usuwania danych oraz do wyszukiwania w bazie wskazanych danych.

W celu wstawienia do istniejącej tabeli nowego wiersza (rekordu) używa się polecenia INSERT, np.:

```
INSERT INTO DaneOsobowe
```

```
values('Kowalski', 12312, 2400)
```

gdzie kolejne wartości w nawiasach okrągłych oznaczają nazwisko, identyfikator i pensję pracownika.

Modyfikacja pamiętanej danej — zwiększenie wynagrodzenia wszystkich pracowników o 5% — ma postać:

```
UPDATE DaneOsobowe
```

```
SET Pensja = Pensja x 1.05
```

a zwiększenie wynagrodzenia tylko wskazanemu pracownikowi ma postać:

```
UPDATE DaneOsobowe
```

```
SET Pensja = Pensja + 500
```

```
WHERE Identyfikator = 1231
```

Usunięcie wskazanej danej — danych wybranego pracownika — ma postać:

```
DELETE FROM DaneOsobowe
```

```
WHERE Identyfikator = 1231
```

Najczęściej używanym jest polecenie SELECT, które służy wyszukiwaniu danych spełniających określone warunki. Wynikiem realizacji polecenia jest zbiór danych. Dane zawarte w tym zbiorze mogą być później pre-

zentowane użytkownikowi w postaci raportów, zestawień lub wykorzystywane w dalszych procesach przetwarzania danych.

Zapytanie sformułowane w SQL polega na wskazaniu tabel bazy danych, w których poszukujemy danych spełniających odpowiedni warunek logiczny. Mechanizmy wyszukiwania opierają się na wykorzystaniu trzech podstawowych operacji wykonywanych na tabelach: selekcji, projekcji i złączenia.

Operacja *selekcji* jest wykonywana na pojedynczej tabeli i oznacza wybór tych wierszy, które spełniają zadany warunek. Np. z tablicy o uproszczonym schemacie (bez podawania typów atrybutów):

Pracownik(Pesel, Imię, Nazwisko, Wiek, Zarobek, DataZatrudnienia)
zapytanie o pracowników, którzy osiągnęli wiek przynajmniej 60 lat i zarabiają powyżej 3 tys. zł, można w języku SQL wyrazić w postaci klauzuli:

```
SELECT * FROM Pracownik WHERE Wiek ≥ 60 AND Zarobek ≥ 3000
```

Symbol * oznacza, że w odpowiedzi otrzymamy zestawy wartości dla wszystkich wyszukanych pracowników. Wynikiem realizacji klauzuli będzie pewna tablica składająca się z tych wierszy tablicy *Pracownik*, które spełniają zadany warunek.

Operacja *projekcji* jest również wykonywana na pojedynczej tabeli i polega na wyborze z niej wskazanych kolumn. Np. z tabeli o podanym wyżej schemacie można utworzyć nową tabelę zawierającą tylko kolumny *Imię* i *Nazwisko*. W języku SQL takie zapytanie wyraża klauzula:

```
SELECT Imię, Nazwisko FROM Pracownik
```

Obie operacje: selekcji i projekcji można połączyć w jednej klauzuli SQL. Np. klauzula:

```
SELECT Pesel FROM Pracownik WHERE Wiek ≥ 60 AND Zarobek ≥ 3000
```

dostarcza listę identyfikatorów tych pracowników, którzy spełniają podany warunek.

Operacja *złączenia* jest wykonywana na dwóch tabelach i polega na utworzeniu nowej tabeli, której wiersze będą połączeniem tych wierszy z dwóch tabel spełniających zadany warunek. Jeden z często spotykanych warunków polega na wskazaniu w jednej tabeli klucza obcego do drugiej tabeli, np. dla tablic o schematach:

Pracownik(Imię, Nazwisko, Zarobek, IdentyfikatorDziału)
Dział(IdentyfikatorDziału, Stanowisko)

odpowiednia klauzula SQL ma postać:

```
SELECT * FROM Pracownik JOIN Dzial  
ON Pracownik.IdentyfikatorDzialu = Dzial.IdentyfikatorDzialu
```

Efektem realizacji tej klauzuli jest tabela o schemacie:

PracownikDzial(Imię, Nazwisko, Zarobek, IdentyfikatorDziału, Stanowisko)

Hurtownie danych

Opisane bazy danych są zwykle wykorzystywane do wspomagania bieżącego funkcjonowania przedsiębiorstw lub organizacji i nazywane *bazami operacyjnymi*. Informacja zgromadzona w bazach operacyjnych jest często zapisem historii funkcjonowania. Analiza tej informacji może służyć przedsiębiorstwom do ich oceny, a także być pomocna przy podejmowaniu decyzji dotyczących zmian ich funkcjonowania. W celu przeprowadzania tego typu analiz wykorzystuje się *hurtownie danych*. Są to wyspecjalizowane bazy danych, które przeprowadzają analizę danych zgromadzonych w bazach operacyjnych, a uzyskane wyniki gromadzą w oddzielnej bazie danych.

Współczesne przedsiębiorstwo, prowadząc działalność produkcyjną, usługową lub handlową, powinno umieć określić swoją pozycję na rynku gospodarczym. Sprowadza się to do odpowiedzi na konkretne pytania (kwerendy), np. w przypadku firmy handlowej:

1. Jak kształtuje się sprzedaż w poszczególnych okresach czasu?
2. Które produkty sprzedają się najlepiej?
3. Jak kształtuje się sprzedaż w rozbiciu na grupy towarowe w poszczególnych okresach czasu z uwzględnieniem konkretnych klientów?
4. Który z oddziałów firmy prosperuje najlepiej, a który najgorzej?
5. W którym mieście sprzedaż jest największa, w którym najmniejsza?
6. Który z klientów kupuje najwięcej, a który najmniej w wybranych okresach czasu?
7. Którzy klienci płacą regularnie, a którzy zalegają z płatnościami, ile zalegają, ile zalegają z uwzględnieniem naliczonych odsetek?

Odpowiedzi na tego rodzaju pytania można udzielić, analizując zawartość operacyjnych baz danych przedsiębiorstwa. Analizy wykorzystania danych zawartych w bazach danych, prowadzone m.in. przez IBM, wskazują, że ich właściciele wykorzystują tylko niecałe 10% danych zgromadzonych w bazach. Powodem jest to, że do właściwego wykorzystania da-

nych często trzeba stosować złożone metody analityczne, co wymaga odpowiedniej wiedzy i jest przeważnie bardzo skomplikowane.

Rozwiązaniem jest stworzenie hurtowni (składnicy) danych, przechowywanych w ujednoliconej i sprawdzonej postaci, niezbędnej do udzielania odpowiedzi na pewną klasę pytań. Hurtownia danych daje wielowymiarowy obraz informacji w niej zawartych. Wymiarami są tu wskazane interesujące przedsiębiorstwo aspekty, wśród nich zwykle występuje wymiar czasowy. Gromadzone w niej dane są informacjami zbiorczymi, których nie można uzyskać bezpośrednio z baz operacyjnych. Istotną cechą hurtowni danych jest to, że dane w niej przechowywane są oddzielone od danych przechowywanych w bazach operacyjnych.

Korzystanie z odpowiednio skonstruowanej hurtowni danych i stworzonych na jej podstawie raportów może pozwalać np. na:

- zwiększenie obrotów dla produktów,
- zahamowanie odpływu klientów mało aktywnych (przez zastosowanie w odpowiednim momencie promocji lub określonej atrakcyjnej oferty),
- zmniejszenie zadłużeń,
- zmniejszenie kosztów marketingowych (np. przez prowadzenie właściwie adresowanych kampanii),
- ograniczenie zbędnych promocji towarów,
- wycofanie lub ograniczenie stanów mało atrakcyjnych towarów.

Najprostszy i najbardziej powszechny narzędziem do tworzenia raportów i kwerend jest pakiet *MS Office*, wyposażony dodatkowo w moduł OLAP (*Online Analytical Processing*), mający wiele zastosowań. Oczywiście istnieją także inne narzędzia programowe do tworzenia raportów na podstawie hurtowni danych, są to narzędzia droższe, ale dające więcej możliwości niż pakiet *MS Office*.

Hurtownie danych można stosować w firmach praktycznie wszystkich działów gospodarki. O potrzebie ich stosowania przekonują statystyki pokazujące, że nawet najmniejsza strata najlepszych klientów naraża firmy na ogromne straty finansowe. Okazuje się, że opieka nad klientami jest jednym z kluczowych zadań firmy (<http://www.odl.com.pl/-top>). Z tego też względu tworzone są *hurtownie danych klientów*. Dysponując danymi o każdym ze swych klientów, można interweniować w odpowiednim momencie, oferować lepsze usługi i produkty, nie dopuszczając tym samym do utraty klienta. Jest to szczególnie ważne w dobie globalnej konkurencji. Dzięki zaawansowanym analizom danych klientów przedsiębior-

stwo może dokonać podziału klientów na grupy, np. o różnej zyskowności. Można w ten sposób zróżnicować formy traktowania potrzeb różnych klientów — przedstawiać dodatkowe oferty dla najlepszych klientów lub ograniczać kontakty z klientami najmniej zyskowymi, redukując w ten sposób koszty ich obsługi. Możliwe jest też prowadzenie ukierunkowanego marketingu, tj. wysyłanie listów tylko do wybranych klientów, planowanie kontaktów, oferowanie „produktów niszowych” dla wybranych, małych grup klientów.

Hurtownie danych stają się istotnym narzędziem wspomagającym działalność firmy. Dzięki analizie danych dziesiątek tysięcy, a nawet setek tysięcy transakcji, dla tysięcy oferowanych produktów, można ściśle określić preferencje klientów i zależności pomiędzy produktami. Umożliwia to wyjście naprzeciw oczekiwaniom klientów i dopasowanie profilu produkcji lub sprzedaży do potrzeb rynku. Działania te przekładają się na wymierne korzyści finansowe, przewyższające koszty wdrożenia hurtowni danych.

IX. ZASTOSOWANIA INFORMATYKI W GOSPODARCE

Wstęp

Informatyka znajduje zastosowanie w różnych dziedzinach nauki, techniki i gospodarki. Szeroko rozumiana gospodarka — przedsiębiorstwa produkcyjne, handlowe i usługowe, a także administracja państwowa i samorządowa — jest jednym z ważniejszych obszarów zastosowań informatyki. Pierwsze zastosowania informatyki na tym polu sięgają lat sześćdziesiątych, ale szerokie upowszechnienie Internetu pod koniec lat osiemdziesiątych dało im nowy impuls. Wspecjalizowane sieci komputerowe były już wprawdzie stosowane wcześniej, np. dysponowały nimi niektóre banki, umożliwiając elektroniczny obrót pieniądza, czy też linie lotnicze, prowadząc rezerwację biletów, ale dopiero Internet stworzył możliwości integracji różnych niezależnie rozwijających się zastosowań.

Wylania się pojęcie *gospodarki elektronicznej*. Jej podstawowym towarem staje *produkt i usługa cyfrowa* — szczególny rodzaj informacji cyfrowej, która ma pewne cechy odróżniające ją od innych rodzajów informacji, takie jak:

- transformowalność — możliwość automatycznego przetwarzania za pomocą komputerów,
- transmitowalność — możliwość przesyłania za pomocą sieci komputerowych,
- replikowalność — nieograniczona możliwość kopiowania,
- niezniszczalność — rozumiana w tym sensie, że nie ulega zmianie w czasie jej przechowywania w pamięci.

Możemy obecnie wyróżnić cztery rodzaje produktów cyfrowych:

- dokumenty — informacje o zaistniałych faktach albo przyjętych zobowiązaniach,
- pieniądze — informacje o instrumentach finansowych, takich jak obligacje, akcje, gotówka,

- utwory autorskie — dzieła literackie, publicystyczne i naukowe, muzyka, filmy,
- oprogramowanie.

Usługa cyfrowa polega na wykorzystaniu produktu cyfrowego do pewnego celu. Np. edytor tekstowy lub system operacyjny są programami — produktami cyfrowymi, natomiast posługiwanie się edytorem lub systemem operacyjnym jest korzystaniem z ich usług.

Funkcjonujące obecnie pojęcie gospodarki elektronicznej (*e-economy*) odnosi się do głównie do:

- przedsiębiorstw,
- współpracy pomiędzy przedsiębiorstwami,
- handlu (*e-commerce*),
- bankowości (*e-banking*).

W gospodarce elektronicznej produkt i usługa cyfrowa są przedmiotem realizacji biznesu, a komunikacja — głównym mechanizmem przekazywania produktu i świadczenia usług.

Z wymianą informacji wiążą się dwa problemy: pierwszy z nich jest natury technicznej, związany z ustaleniem jednolitej formy przekazywanej informacji, tak by mogła być ona odczytywana i jednoznacznie interpretowana przez odbiorców. Pomocą służy tu opracowany przez ISO pod koniec lat osiemdziesiątych standard elektronicznej wymiany EDI (*Electronic Data Interchange*). Drugi problem ma inny charakter i jest związany z prowadzeniem negocjacji w celu ustanowienia warunków współpracy; jest on jednocześnie jądrem biznesu elektronicznego.

Nowe możliwości komunikowania się stały się podstawą nowych form świadczenia pracy — *telepracy* oraz uczenia się — *zdalnego nauczania* (*e-learning*).

Telepraca polega na wykonywaniu pracy w miejscu zamieszkania, bez potrzeby fizycznej obecności w siedzibie przedsiębiorstwa. Umożliwia to m.in. tworzenie rozproszonych zespołów pracowników, którzy zlokalizowani w różnych miejscach uczestniczą we wspólnych projektach, mając wspólny dostęp do tej samej przestrzeni wirtualnej, widząc nawzajem wyniki swojej pracy i komunikując się na bieżąco. Telepraca w wielu krajach, w tym w Polsce, stanowi nową formę świadczenia pracy, ciągle jeszcze nieuznaną formalnie przez prawo. Krajem europejskim, w którym telepraca jest najbardziej powszechna, jest Irlandia.

Zdalne nauczanie jest formą zdobywania wiedzy w miejscu zamieszkania. Jest to forma ciągle jeszcze niedoskonała, gdyż nakłada duże wy-

mogi komunikacyjne — niezbędna jest komunikacja multimedialna, a ponadto wymaga ogromnej pracy przy przygotowaniu materiałów dydaktycznych. Zdalne nauczanie ma już pewne zastosowanie w tych krajach, w których warunki naturalne — rozproszenie i duże odległości, czynią je naturalną formą, np. nauka na poziomie szkoły podstawowej w Australii.

Informatyka w przedsiębiorstwach

Mówiąc o zastosowaniu informatyki w przedsiębiorstwach, mamy na myśli systemy informatyczne wspomagające produkcję, gdyż inne systemy, również w przedsiębiorstwach, np. służące do wspomagania funkcji finansowo-księgowych czy też prowadzenia spraw osobowych, są stosowane w zasadzie we wszystkich instytucjach i organizacjach.

Pierwsze systemy wspomagania produkcji powstały w latach sześćdziesiątych i służyły do kontroli stanów magazynowych, w latach siedemdziesiątych na ich podstawie zostały stworzone systemy planowania zapasów MRP (*Material Requirement Planning*). Systemy MRP pozwoliły na kontrolę przepływu surowców i komponentów, umożliwiając planowanie zapasów z takim wyprzedzeniem, by zapewnić płynność procesów produkcyjnych.

W latach dziewięćdziesiątych nastąpił dalszy rozwój, przynosząc tzw. zintegrowane systemy zarządzania przedsiębiorstwem ERP (*Enterprise Resource Planning*). Systemy tej klasy służą do kompleksowego zarządzania przedsiębiorstwem, obejmując obsługę i planowanie finansów, zarządzanie zasobami ludzkimi, zakupy, zarządzanie zapasami, planowanie produkcji. Podstawą do zarządzania w obrębie poszczególnych dziedzin jest wspólna baza danych gromadząca wszystkie istotne fakty dotyczące funkcjonowania przedsiębiorstwa. Ze względu na nowe możliwości współpracy przedsiębiorstwa ze swoimi dostawcami i odbiorcami za pośrednictwem Internetu pojawiły się rozszerzone systemy zintegrowanego zarządzania EEA (*Extended Enterprise Applications*).

Obecnie systemy wspomagania produkcją wkroczyły w nową fazę rozwoju, obejmując sieci przedsiębiorstw uczestniczących w realizacji wspólnego produktu. Potrzeba taka wyłoniła się w naturalny sposób, gdyż wytwarzanie coraz większej liczby finalnych produktów rynkowych, takich jak: samochody, samoloty, a nawet urządzenia gospodarstwa domowego — pralki czy lodówki, opiera się na współpracy wielu firm specjalizujących się w produkcji podzespołów. Nowa, wyłaniająca się kategoria systemów to zarządzanie łańcuchami dostaw SCM (*Supply Chain Mana-*

gement). Istotną częścią tych systemów, poza oczywistą potrzebą zarządzania produkcją w poszczególnych współpracujących ze sobą przedsiębiorstwach, jest elektroniczna wymiana informacji. Wymiana ta jest potrzebna zarówno przed utworzeniem łańcucha współpracujących przedsiębiorstw, jak i w trakcie procesu wytwarzania produktów.

Systemy ERP współdziałają coraz częściej z innym rodzajem oprogramowania, który służy do współpracy z klientami (dostawcami i odbiorcami produktów i usług). Chodzi tu o tzw. systemy zarządzania kontaktami z użytkownikami CRM (*Customer Relationship Management*). Systemy CRM mają służyć optymalizacji funkcjonowania przedsiębiorstwa przez ukierunkowanie na klienta. Chodzi o takie zaspokajanie potrzeb indywidualnych klientów, by wiązać ich z firmą na możliwie długi czas. Wyróżnia się trzy kategorie systemów CRM: operacyjny, analityczny i kontaktowy.

Typowe funkcje operacyjnego systemu CRM obejmują obsługę klienta, zarządzanie zamówieniami, wystawianie rachunków, a także komunikację z klientami; są wykorzystywane m.in. w działach sprzedaży, serwisu i marketingu.

Analityczny system CRM pozyskuje, przechowuje i analizuje dane o klientach. Celem analizy jest zrozumienie potrzeb i zachowań klientów, tak by było możliwe oszacowanie ryzyka planowanych działań przedsiębiorstwa. Funkcjonowanie analitycznego systemu CRM wiąże się z istnieniem hurtowni danych.

Kontaktowy system CRM jest zintegrowanym centrum komunikacyjnym, który za pomocą różnych środków komunikacji — głosowej, telefonicznej, SMS-owej, faksowej, poczty elektronicznej, stron i portali internetowych, a nawet tradycyjnej poczty — utrzymuje kontakt z partnerami przedsiębiorstwa (klientami, dostawcami, kooperantami).

Systemy CRM funkcjonujące w przedsiębiorstwach zawierają zwykle wszystkie wymienione kategorie systemów, przy czym spośród możliwych ich funkcji są wybierane tylko te, które w danych warunkach są najbardziej odpowiednie.

Informatyka w bankach

Rola banków w gospodarce jest znana od dawna, tempo rozwoju gospodarczego jest silnie uwarunkowane sprawnym obiegiem pieniądza. Internet stwarza bankom możliwość świadczenia nowych form usług. Do usług tych należą m.in.:

- internetowe serwisy informacyjne, przedstawiające podstawowe informacje o banku, jego strukturze, usługach;
- zdalny dostęp do własnych rachunków bankowych z komputera domowego (*homebanking*) za pośrednictwem sieci telefonicznej; bank oferuje odpowiednie oprogramowanie, które umożliwia dostęp i jednocześnie chroni przez dostępem nieupoważnione osoby; coraz większa liczba banków oferuje tę usługę także za pośrednictwem telefonu komórkowego;
- bankowość internetowa, która pozwala na dyspozycje przelewu, zakładanie i likwidację lokat, polecenia zapłaty, przeglądanie historii rachunku itd.;
- elektroniczne karty płatnicze pozwalające na realizację rachunków i pobieranie gotówki z bankomatów.

W Polsce od końca lat dziewięćdziesiątych usługi bankowe za pośrednictwem Internetu oferują wszystkie większe banki. Szczególnie istotne przy korzystaniu z internetowych usług bankowych jest uwierzytelnienie klienta i bezpieczeństwo wymienianych danych. Istnieją tu różne rozwiązania, sprzętowe i programowe, np. karty montowane do komputera przeznaczonego do szyfrowania komunikacji z bankiem lub podpisy elektroniczne do uwierzytelniania klienta.

Usługi i handel elektroniczny

Istotą biznesu elektronicznego jest doprowadzenie do kontraktu dotyczącego kupna-sprzedaży towarów lub usług pomiędzy najczęściej dwiema stronami: sprzedającym i kupującym. Firmy działające za pośrednictwem Internetu opierają swoją współpracę najczęściej na jednym z dwóch modeli:

- model brokerski — polega na usługach pośrednictwa w doprowadzeniu do spotkania kupującego i sprzedającego;
- model promocyjny — polega na darmowej ofercie produktów lub usług w zamian za możliwość przesłania klientowi materiałów promocyjnych.

Oprócz nich wyróżniamy:

- model handlowca — reprezentuje sprzedawcę detalicznego lub hurtowego, który sprzedaje towary na podstawie cen katalogowych lub przez aukcje;

- model abonencki — polega na dostępie do zawartości witryny internetowej za ustaloną opłatą abonentową.

Nowym zjawiskiem związanym z zastosowaniem informatyki jest zdalne świadczenie różnego rodzaju usług za pośrednictwem Internetu, które określa się jako *outsourcing*. Termin ten nie ma dotąd utrwalonego odpowiednika w języku polskim, można próbować go tłumaczyć jako internetowe podwykonawstwo usług. Zakres świadczonych zdalnie usług jest szeroki, może dotyczyć problemów ściśle technicznych, np. zdalne administrowanie lokalną siecią komputerową przedsiębiorstwa, bądź też usług aplikacyjnych, np. prowadzenie finansów i księgowości czy zarządzanie zasobami ludzkimi.

Z zastosowaniami informatyki w gospodarce i w administracji wiąże się ważna, niestety ciągle aktualna, uwaga. Często można spotkać się z poglądem, że wdrożenie systemów informatycznych sprowadza się do zakupu i zainstalowania odpowiedniego oprogramowania. Tak jednak nie jest! Informatyzacja jest procesem bardzo złożonym, stanowi bowiem fragment ewolucyjnego procesu zmian w przedsiębiorstwie. Samo wdrożenie systemu informatycznego jest tylko częścią, często drobną, przedsięwzięcia, które trzeba zrealizować, aby informatyzacja przyniosła jakiegokolwiek korzyści. Należy ponadto przypomnieć, że przez długi okres informatyzację rozumiano jeszcze prymitywniej, utożsamiając ją z komputeryzacją, czyli z wyposażeniem przedsiębiorstwa lub instytucji w pewną partię sprzętu komputerowego. Poglądy takie zapisały się w historii licznymi nieudanymi projektami.

Spółeczeństwo informacyjne

Koniec ubiegłego wieku zapoczątkował nowy okres cywilizacyjnego rozwoju ludzkości nazywany okresem informacyjnym, odróżniając go w ten sposób od okresów wcześniejszych: agrarnego i przemysłowego. Informatyka ma tu znaczący, ale oczywiście nie jedyny wpływ — jest ona elementem łańcucha wielu wzajemnie sprzężonych dziedzin. Swoją rolę informatyka zawdzięcza ogromnemu postępowi, zwłaszcza technologii elektronicznych, a rosnące dzięki temu możliwości środków obliczeniowych umożliwiają kolejne usprawnienia, m.in. w technologii elektronicznej, a także w poszukiwaniu innych technologii, które mogą mieć znaczenie dla konstrukcji komputerów.

Technologie informatyczne mają duży wpływ na różne dziedziny, m.in. harmonijny czy zrównoważony rozwój społeczeństw (*sustainable*

development). Pojęcie to oznacza proces poszerzania ludzkich wyborów i wykorzystania zdolności przez kształtowanie kapitału społecznego, tak by w możliwie pełny sposób zaspokoić potrzeby obecnych generacji, bez narażania na szwank potrzeb przyszłych pokoleń.

Np. w zakresie ochrony środowiska naturalnego proces ten polega na:

- przyspieszaniu odchodzenia od technologii surowcowochołnych,
- skutecznym badaniu zagrożeń związanych ze składowaniem odpadów, pustynnienia i zmianami klimatycznymi,

w zakresie ekonomii polega on na:

- restrukturyzacji gospodarki przez zastępowanie surowcowochołnych technologii produkcyjnych przez zasobooszczędne technologie,
- zwiększeniu efektywności regulacji rynkowych przez wzbogacanie oferty dóbr i usług, redukcję kosztów transakcyjnych, ułatwienie kontroli procesów produkcji, dystrybucji i konsumpcji,

w zakresie polityki polega na:

- wzmocnieniu administracji państwowej, dając jej narzędzia centralnego planowania w odniesieniu do wybranych dziedzin, np. zasobów naturalnych,
- stwarzaniu szans międzynarodowej solidarności i porozumienia dzięki zagęszczonej sieci komunikacyjnej,

w zakresie społeczeństwa polega na:

- obniżeniu psychologicznej bariery obcości względem ludzi w innych krajach,
- wzmocnieniu międzynarodowych systemów regulacji, zwiększając ich zdolności ustanawiania norm i reguł w celu kontrolowania nadużyć w zakresie praw człowieka, eksploataowania zasobów, ochrony środowiska naturalnego.

Świadomość wpływu technologii informatycznych na rozwój cywilizacyjny stała się przyczyną wyodrębnienia pojęcia *społeczeństwa informacyjnego*. Należy je traktować raczej jako realną wizję, ku której będzie zmierzać społeczny rozwój, aniżeli stwierdzenie faktycznego stanu. Oferując nowe możliwości, wizja ta niesie też nowe wyzwania. Ich źródłami są dwa techniczne wynalazki — komputer i telekomunikacja, które stanowią jądro technologii informatycznych.

Zdolności obliczeniowe komputera przesądzają, że każda intelektualna czynność, którą można z góry zaplanować, będzie wykonywana lepiej

przez komputer niż przez człowieka. A rosnące możliwości telekomunikacyjne powodują przyspieszenie procesów globalizacyjnych.

Obecny okres jest czasem transformacji społeczeństw i dotyczy zarówno pojedynczych ludzi i grup zawodowych, jak i całych krajów. Jego istotą jest zdobywanie przez ludzi odpowiedniego wykształcenia, tak by skorzystać z potencjalnych możliwości i uniknąć problemu wykluczenia ze społeczeństwa.

X. PRZEDSIĘWZIĘCIA INFORMATYZACYJNE³

Informatyzacja

Informatyzacja polega na wykorzystaniu środków informatycznych — sprzętu i oprogramowania — do realizacji obiegu i przetwarzania informacji w pewnym systemie informacyjnym. Mówiąc o systemie informacyjnym przedsiębiorstwa lub instytucji, abstrahujemy od formy informacji (czy są to przekazy ustne, dokumenty papierowe czy elektroniczne) i od środków użytych do jej przetwarzania (czy są to ludzie, czy urządzenia mechaniczne bądź elektroniczne). Informatyzacji podlega zwykle tylko część systemu informacyjnego — ta, w której przetwarzanie informacji odbywa się za pomocą środków informatycznych. Podstawą informatyzacji jest zatem identyfikacja tego fragmentu systemu informacyjnego, który ma być objęty informatyzacją. Informatyzacja może mieć różny zakres — od pojedynczego stanowiska pracy do całego przedsiębiorstwa.

W zależności od skali informatyzacja przybiera różne formy. W przypadku pojedynczego stanowiska pracy może sprowadzić się do wyposażenia lub doposażenia go w komputer wraz z typowym oprogramowaniem. Tak jest np. przy tworzeniu typowych stanowisk pracy biurowej. Wykorzystywane oprogramowanie jest tu zwykle oprogramowaniem „z półki”, to znaczy jest ono kupowane tak jak każdy produkt masowy, czyli nie jest „dedykowane” konkretnemu użytkownikowi. Natomiast w przypadku przedsiębiorstwa informatyzacja wymaga zupełnie innego podejścia, opartego na sformułowaniu i realizacji projektu informatycznego.

Projektem informatycznym jest każda planowa działalność, obejmująca zestaw zadań mających wspólny cel — wytworzenie systemu informatycznego lub produktu programowego. W pierwszym przypadku chodzi o wytworzenie lub adaptację istniejącego oprogramowania do zaspokojenia

³ Rozdział ma charakter uzupełniający, jego celem jest uświadomienie problemów, z którymi mogą zetknąć się osoby, które nie są informatykami, podczas ich zaangażowania w przedsięwzięcia informatyzacyjne.

nia potrzeb konkretnego klienta — przedsiębiorstwa lub instytucji, w drugim — o wytworzenie produktu, który można oferować szerokiej klasie odbiorców.

Wspólny cel wyraża zgodność interesów, dążeń i działań w obrębie dobrze wyodrębnionej jednostki — przedsiębiorstwa, organizacji gospodarczej. Takimi celami są m.in.:

- zastąpienie dotychczasowego niewydolnego systemu nowym systemem informatycznym — zwłaszcza gdy przetwarza się dane dużej liczby klientów;
- obniżka kosztów eksploatacji istniejącego systemu — ceny sprzętu komputerowego spadają, wzrastają jego możliwości obliczeniowe, a masowo używane oprogramowanie ma szerokie zastosowanie;
- lepszy — szybszy i pełniejszy obieg informacji, co jest niezwykle istotne przy podejmowaniu decyzji i w pracy zespołowej;
- konkurencyjność w obsłudze klientów — przejrzystość usług, ale i ich rozpowszechnienie i skrócenie czasu dostępu, jak to ma miejsce w przypadku bankomatów czy wykorzystania Internetu;
- modyfikacja istniejącego systemu w związku ze zmianami legislacyjnymi, np. wprowadzenie ochrony danych osobowych, zmiana przepisów podatkowych, celnych.

Zestaw zadań wskazuje na złożoność przedsięwzięcia, na możliwość jego dekompozycji na mniejsze jednostki. W projekcie informatycznym występują zadania techniczne i organizacyjno-menedżerskie. Zadania techniczne koncentrują się na precyzyjnym sformułowaniu wymagań i wytworzeniu oczekiwanego produktu programowego. Zadania organizacyjno-menedżerskie wiążą się z inicjowaniem przedsięwzięcia, oszacowaniem kosztu, doбором wykonawców, określeniem harmonogramu prac i zapewnieniem infrastruktury niezbędnej do realizacji przedsięwzięcia.

Planowa działalność wskazuje na konieczność koordynacji zadań, ich części składowych i komunikacji pomiędzy uczestnikami projektu.

Typowa struktura realizacji projektu informatycznego wyróżnia następujące fazy:

1. *Planowanie strategiczne*. Punktem wyjścia jest ocena sytuacji w organizacji, która pozwala rozważyć podjęcie projektu informatycznego i związanych z tym kosztów. Organizacja musi bowiem nadać swoim potrzebom odpowiednie priorytety. Wymaga to identyfikacji celów biznesowych projektu, harmonogramu i budżetu osiągania tych celów, wskazania osób zarządzających i odpowiedzialnych za jego realizację.

2. *Formułowanie założeń.* Założenia przedstawiają podstawowe informacje o zamierzonym przedsięwzięciu: cele biznesowe, wizję systemu (w tym zadania użytkowników, założenia jakościowe, parametry techniczne, zgodność ze standardami, organizację i działanie systemu), podstawowe wymagania i ograniczenia (np. ograniczenia zasobowe, powiązane projekty, wymagania i ograniczenia prawne). Założenia mają na celu przygotowanie decyzji o podjęciu prac nad wytworzeniem docelowego produktu.

3. *Analiza wykonalności.* Na podstawie założeń dokonuje się oceny możliwości realizacji projektu z różnych punktów widzenia:

- wykonalności technicznej, obejmującej studium funkcji, efektywności i ograniczeń, które mogą mieć wpływ na zdolność osiągnięcia akceptowanego systemu;
- wykonalności organizacyjnej, obejmującej analizę zmian struktury organizacji, zmian w pracy ludzi, wskazania pożądanych nowych umiejętności;
- wykonalności ekonomicznej, skupiającej się na szacunku kosztów konstrukcji i spodziewanych zysków w perspektywie krótko- i długofalowej;
- wykonalności prawnej, oceniającej zgodność wykonywanych funkcji z istniejącym stanem prawnym lub postulowanymi zmianami legislacyjnymi.

4. *Opis i ustanowienie projektu.* Opis projektu zawiera jego wstępny plan, którego istotnymi elementami są ramowy harmonogram realizacji oraz miary oceny i kryteria akceptacji systemu. Zasadniczym celem wstępnego planu jest uporządkowane zestawienie informacji niezbędnych do podjęcia decyzji o ustanowieniu projektu. Z kolei ustanowienie projektu obejmuje:

- uzyskanie formalnej aprobaty (zarządu, dyrekcji lub właściciela) organizacji,
- przydzielenie budżetu,
- zdefiniowanie struktury projektu,
- usytuowanie struktury zespołu wykonawców, w tym kierownika projektu, wskazanie jego zwierzchnika, ustalenie składu komitetu sterującego i przedstawicieli użytkownika, administracji projektu,
- przyjęcie zasad zarządzania i raportowania,
- opublikowanie celów i planu projektu.

5. *Planowanie projektu*. Obejmuje ono zarówno zadania techniczne, jak i menedżerskie. Szczegółowy plan projektu zwykle zawiera:

- omówienie projektu: oczekiwane rezultaty, odniesienia i definicje pojęć,
- organizację projektu: model procesu, strukturę organizacyjną, granice i interfejsy, przypisane odpowiedzialności,
- proces zarządzania: cele i priorytety zarządzania, uwarunkowania i ograniczenia, zasady zarządzania ryzykiem, monitorowania i nadzoru, organizacji zespołu,
- zagadnienia techniczne: metody, narzędzia i technologie, zasady dokumentowania oprogramowania, wsparcie projektu,
- zadania, harmonogram, sposoby zapewniania jakości.

6. *Konstruowanie*. Zasadnicze są tu zadania związane z techniczną realizacją systemu, będące domeną specjalistów informatyków. Sens poprzednich etapów sprowadza się do tego, by w sposób jednoznaczny i wyczerpujący określić to, co mają zrobić informatycy i dzięki temu stworzyć podstawę do zawarcia i rozliczenia kontraktu pomiędzy zamawiającym (inwestorem) a wykonawcą (firmą informatyczną).

7. *Weryfikacja*. Etap ten następuje bezpośrednio po zrealizowaniu systemu. Jego celem jest sprawdzenie, po pierwsze, czy system został zrealizowany zgodnie z założeniami kontraktu, po drugie, czy odpowiada on faktycznym potrzebom instytucji. Należy zwrócić uwagę, że te dwa cele są różne. Stwierdzenie, że system został zrealizowany zgodnie z założeniami zawartymi w kontrakcie, nie musi oznaczać, że „pasuje” on do potrzeb instytucji, która ma być jego użytkownikiem. Oznacza to tylko, że źle opracowano założenia albo warunki funkcjonowania instytucji uległy istotnej zmianie w czasie realizacji projektu i że trzeba ponieść niepotrzebne, przynajmniej częściowo, koszty (w przypadku niektórych kategorii oprogramowania do tego etapu można zaliczyć dodatkowo certyfikację, czyli sprawdzenie, czy jego funkcjonowanie, budowa i dokumentacja spełniają ogólne wymagania odnoszące się do danej kategorii).

8. *Wdrożenie*. Etap rozpoczyna się zwykle wstępną eksploatacją nowego, zweryfikowanego systemu, czasem równoległe ze starym systemem. Towarzyszą temu inne działania, np. szkolenie załogi, zmiany organizacji pracy instytucji. Etap kończy się z chwilą całkowitego zastąpienia starego systemu nowym.

9. *Eksploatacja*. Etap ten można potraktować jako odrębną część, poza zasadniczym projektem informatycznym. Należy jednak pamiętać, że

w okresie poprawnej eksploatacji systemu mogą ujawnić się wcześniej niewykryte usterki oraz pojawić się potrzeby drobnych korekt — wszystkie te sytuacje wymagają działań określanych mianem konserwacji lub pielęgnacji oprogramowania, zwykle realizowanej przez wykonawcę oprogramowania na podstawie oddzielnie zawartego kontraktu.

Przedstawiony schemat wskazuje na stopień zaangażowania nieinformatyków w projekt informatyczny. To specjaliści w danej dziedzinie zastosowań — reprezentanci branży, w której jest zaangażowana informatyzowana instytucja, a nie informatycy, precyzują informatykom cele do realizacji i dokonują oceny poprawności ich realizacji.

Cykl życia oprogramowania

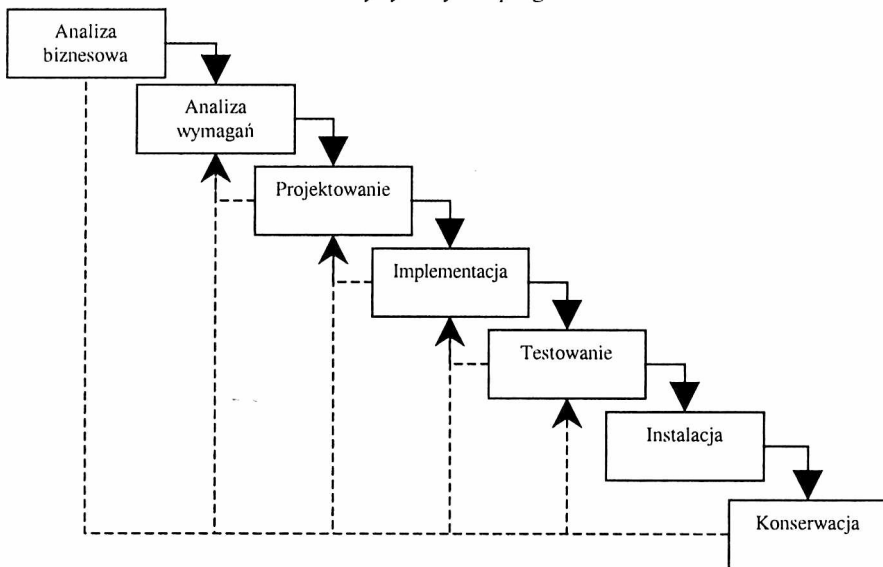
W celu uświadomienia konsekwencji realizacji projektów informatycznych przedstawimy *cykl życia oprogramowania*, czyli zestaw czynności związanych z wytwarzaniem i użytkowaniem oprogramowania — od momentu podjęcia decyzji o jego wytworzeniu, przez konstrukcję, wdrożenie, eksploatację, aż do wycofania z użycia. Wytworzenie oprogramowania jest zwykle zasadniczą częścią projektu informatycznego.

Tradycyjny model cyklu życia oprogramowania jest nazywany modelem *kaskadowym* albo *wodospadowym*. Wyróżnia on następujące fazy życia oprogramowania:

- faza analizy biznesowej,
- faza określania wymagań,
- faza projektowania,
- faza implementacji albo kodowania,
- faza testowania,
- faza instalacji,
- faza konserwacji (utrzymania).

Fazy te w zasadzie występują po sobie kolejno; wyniki uzyskane w jednej fazie są podstawą do prac w fazie następnej. Ocena wyników (artefaktów) danej fazy, poza fazą analizy wymagań, polega ich porównaniu z wynikami (artefaktami) fazy poprzedzającej. W praktyce często fazy nakładają się na siebie, a dodatkowo dopuszcza się powroty z danej fazy do faz wcześniejszych w celu poprawek, uściśleń lub uzupełnień wcześniej opracowanych dokumentów. Przyczyną powrotu może być wykrycie niespójności, niejednoznaczności lub braku określoności w artefaktach poprzednich faz.

Rys. 10.1
Model kaskadowy cyklu życia oprogramowania



Model kaskadowy wprowadza nie tylko pewną kolejność wykonywania różnych czynności, ale także określa ramy organizacyjne związane z wyznaczaniem zespołów, przydzielania im odpowiednich zasobów i rozliczania efektów prac. Model jest podstawą do wyróżnienia różnych kategorii specjalistów zaangażowanych w budowę i funkcjonowanie systemów informatycznych, np. analityków, projektantów, programistów, inspektorów, administratorów czy konserwatorów.

Analiza biznesowa wiąże się z miejscem — organizacją lub instytucją, w którym ma powstać system informatyczny. System taki angażuje ludzi, sprzęt komputerowy i oprogramowania, realizując pewien obieg i przetwarzanie danych. Celem analizy jest ustalenie ogólnych oczekiwań w stosunku do mającego powstać systemu i wskazanie tych jego części, w których przetwarzanie danych będzie najlepiej realizowane przez oprogramowanie i sprzęt, a także tych, które będą realizowane przez ludzi współpracujących z powstałym systemem.

Analiza wymagań wiąże się z dziedziną, w której pojawiają się problemy wymagające wprowadzenia systemu informatycznego. Polega na rozpoznaniu tych wszystkich aspektów rzeczywistości, które mogą mieć wpływ na postać oczekiwanego systemu lub na sposób jego budowy, wdrożenia

bądź funkcjonowania. Celem jest określenie wymagań użytkownika systemu informatycznego.

Wyróżnia się wymagania funkcjonalne i niefunkcjonalne. Wymagania funkcjonalne określają zakres wykonywanych funkcji (usług) systemu, a wymagania niefunkcjonalne mogą odnosić się do jakości systemu lub do procesu wytwarzania systemu.

Można wyróżnić dwie podstawowe kategorie wymagań jakościowych: wydajnościowe i wiarygodnościowe.

Do wymagań *wydajnościowych* można zaliczyć: przepustowość systemu, czasy reakcji i czasy obsługi użytkowników. Np. przy budowie systemów bankowych należy określić liczbę przewidywanych do obsługi klientów, a także czas realizacji zlecanych przez nich transakcji.

Na wymagania *wiarygodnościowe* składają się:

- *dyspozycyjność* — jest miarą gotowości systemu do użycia; przykładem systemu o ostrych wymogach dyspozycyjności jest system sterowania reaktorami w elektrowniach jądrowych czy system sterowania międzynarodowym lub międzymiastowym ruchem telekomunikacyjnym; systemy takie powinny pracować całą dobę bez przerw, przez wszystkie dni roku;
- *niezawodność* — określa odpowiednio długie okresy czasu bezawaryjnej pracy; bezpośrednio nie można tego pojęcia odnosić do oprogramowania, gdyż może być ono tylko poprawne albo niepoprawne, ale oprogramowanie funkcjonujące w zawodnym środowisku wykonawczym powinno przewidywać awarie tego środowiska i reagować tak, by zachować poprawność wykonywanych funkcji; przykładem takiego systemu jest system sterowania rakietami kosmicznymi;
- *bezpieczeństwo* — wiąże się w pewnym zakresie z wymogiem niezawodności; oznacza odpowiednie reagowanie na awarie, przy czym nie chodzi o awarie środowiska wykonawczego, ale o awarie w otoczeniu systemu; w takim przypadku system powinien podejmować działania specyficzne, odbiegające od rutynowych; przykładem jest system sterowania reaktorem atomowym, który w przypadku stwierdzenia zajścia krytycznych lub niedopuszczalnych zjawisk w reaktorze powinien rozpocząć procedurę awaryjnego wyłączenia reaktora; ogólnie, wymóg bezpieczeństwa odnosi się do tych systemów, których funkcjonowanie jest związane z bezpieczeństwem ludzi lub środowiska;

- *ochrona (zabezpieczenie)* systemów — wiąże się z zapobieganiem nieupoważnionemu dostępowi i manipulacji na danych pamiętanych w systemie; przykładem są systemy bankowe lub osobowe.

Poza wymogami wydajnościowymi i wiarygodnościowymi mogą być formułowane inne wymogi dotyczące jakości produktu. Ważnym wymogiem jest np. skalowalność. Oznacza ona, że w przypadku rozbudowy systemu zachowuje poprzednio wymienione wymogi. Skalowalność może być pełna lub częściowa, w zależności od tego, czy żąda się zachowania wszystkich, czy tylko niektórych wymogów. Np. można żądać, by w przypadku zwiększenia liczby użytkowników system gwarantował stałe czasy obsługi, zachowywał dyspozycyjność i niezawodność.

Wymagania odnoszące się do procesu wytwarzania systemu można podzielić na dwie kategorie: organizacyjno-menedżerskie i techniczne. Wymagania organizacyjno-menedżerskie zawsze wiążą się z ograniczeniami terminowymi (terminy wykonania etapów, termin zakończenia całości, miejsca i terminy wdrożeń), finansowymi lub kadrowymi. Wymagania techniczne mogą dotyczyć korzystania ze wskazanych platform sprzętowo-programowych, prowadzenia projektu zgodnie z ustaloną metodyką, posługiwania się wskazanymi narzędziami wspomagającymi proces wytwarzania.

Charakteryzując wymagania нефunkcjonalne, można stwierdzić, że pierwsza grupa wymagań, odnoszących się do jakości systemu, wskazuje na pewien cel, podczas gdy druga grupa wymagań, odnoszących się do procesu wytwarzania systemu, wprowadza pewne ograniczenia w postępowaniu prowadzącym do uzyskania tego celu.

Z punktu widzenia organizacji zamierzającej wprowadzić system informatyczny analizę wymagań kojarzy się ze studium osiągalności, którego celem jest stwierdzenie, czy przy założonych środkach i ograniczeniach możliwe jest osiągnięcie zamierzonego celu.

Projektowanie polega w pierwszej kolejności na określeniu architektury systemu informatycznego, czyli składowych systemu i wzajemnego ich powiązania. Wyróżnia się składowe sprzętowe — węzły (np. komputery lub urządzenia wejścia-wyjścia) i składowe informacyjne — komponenty (np. programy, bazy danych, biblioteki programowe), które są ulokowane w węzłach. Powiązania między węzłami mogą być realizowane przez pojedyncze łącza lub sieci komunikacyjne. Powiązania między komponentami są bardziej złożone, ich głównym aspektem jest ustalenie zakresu wzajemnie świadczonych sobie usług. W następnej kolejności projektowanie polega na szczegółowej specyfikacji poszczególnych składo-

wych architektury. Powinna ona jednoznacznie określić rodzaj wymaganego sprzętu i funkcje poszczególnych komponentów oprogramowania z taką dokładnością, by programista mógł napisać odpowiedni kod w wybranym języku programowania. Celem projektowania jest więc dobór odpowiednich środków informatycznych i zorganizowanie ich w taki sposób, aby zaprojektowany system spełniał wymagania użytkownika.

Implementacja jest domeną ściśle informatyczną. Wyróżnia się tu dwa rodzaje czynności: kodowanie i testowanie. Kodowanie polega na napisaniu tekstu programów w wybranych językach programowania, a testowanie — na eksperymentalnym sprawdzeniu, czy system działa poprawnie. Wyróżnia się przy tym dwa rodzaje testowania:

- testowanie jednostkowe — polegające na oddzielnym sprawdzeniu poprawności poszczególnych komponentów programowych,
- testowanie integracyjne — polegające na sprawdzeniu poprawności współpracy wszystkich komponentów.

Testowanie jako kolejna faza ma na celu sprawdzenie, czy zbudowany system działa poprawnie w środowisku użytkownika, czy rzeczywiście spełnia jego oczekiwania. Podczas gdy w testowaniu w poprzedniej fazie chodziło o stwierdzenie, czy system działa zgodnie ze specyfikacją precyzyjnie ustaloną podczas projektowania, to w obecnej fazie chodzi o stwierdzenie spełnienia oczekiwań użytkownika, które, w odróżnieniu od specyfikacji projektowych, być może nie zostały całkowicie precyzyjnie ustalone w wymaganiach użytkownika. Testowanie jako oddzielna faza ma charakter walidacyjny, podczas gdy testowanie w fazie implementacyjnej ma charakter weryfikacyjny.

Faza instalacji wiąże się z wdrożeniem zrealizowanego i przetestowanego systemu w środowisku użytkownika. Wdrażanie nowego systemu może polegać na okresowym jednoczesnym funkcjonowaniu systemu starego i nowego. Jest to strategia kosztowna, ale bezpieczna dla funkcjonującej instytucji. Możliwe są oczywiście inne strategie, lecz z ich zastosowaniem wiąże się potrzeba oszacowania ryzyka niepowodzenia przedsięwzięcia.

Faza konserwacji (utrzymania) jest związana z eksploatacją oprogramowania. Wykonywanymi tu czynnościami są generacja i instalacja oprogramowania w konkretnym środowisku wykonawczym, rekonfiguracja oprogramowania powodowana zmieniającymi się warunkami eksploatacji, usuwanie zauważonych błędów, a także modyfikacje oprogramowania na skutek zmieniających się wymogów użytkowych. W przypadku istot-

nych modyfikacji należy dokonać powrotu aż do fazy analizy. Właśnie ten fakt jest zasadniczą wadą modelu, gdyż błędy poczynione na etapie analizy wymagań powodują duże straty finansowe.

Cztery pierwsze fazy cyklu życia oprogramowania — analiza wymagań, projektowanie, implementacja i testowanie — określa się często jako cykl wytwarzania oprogramowania.



LITERATURA

- Brzeziński M. (red.), *Organizacja i sterowanie produkcją. Projektowanie systemów produkcyjnych i procesów sterowania produkcją*, Agencja Wydawnicza Placet, Warszawa 2002.
- Cellary W. (red.), *Polska w drodze do społeczeństwa informacyjnego. Raport o rozwoju społecznym*, UNDP, Warszawa 2002.
- Grzechnik J., *Bankowość internetowa*, Internetowe Centrum Promocji, Gdańsk 2000.
- Harel D., *Rzecz o istocie informatyki. Algorytmika*, Wydawnictwa Naukowo-Techniczne, Warszawa 1992.
- Majewski J., *Informatyka dla logistyki*, Instytut Logistyki i Magazynowania, Poznań 2002.
- Metzger P., *Anatomia PC. Kompendium*, Wydawnictwo Helion, Gliwice 2002.
- Szejko S. (red.), *Metody wytwarzania oprogramowania*, Wydawnictwo Mikom, Warszawa 2002.
- Whitehorn M., Marklyn B., *Relacyjne bazy danych*, Wydawnictwo Helion, Gliwice 2002.

WYDAWNICTWO FORUM NAUKOWE

Zadaniem Wydawnictwa FORUM NAUKOWE jest realizacja następujących celów:

- dostarczenie studentom informacji naukowej i dydaktycznej,
 - możliwość wymiany myśli naukowej w obrębie Uczelni i poza nią,
 - możliwość realizacji ambicji naukowych przez pracowników Uczelni w zakresie publikacji naukowych.
-

W Wydawnictwie ukazały się m.in. następujące publikacje:

Chelpa S., *Negocjacje w biznesie. Zagadnienia kluczowe*
Dubisz J., Olejniczak Z., *Inwestycje kapitałowe. Ryzyko, wycena i dywersyfikacja portfeli inwestycyjnych*
Europa w Polsce – Polska w Europie, część I-III
Jendrośka J., *Postępowanie administracyjne i sądowno-administracyjne*
Kapusta F., *Logistyka*
Kraśniński Z., *Rynek usług uzdrowiskowych w Polsce*
Krzykała F., *Główne problemy socjologii gospodarczej, część I-III*
Labuda G., *Rozwój metod dziejopisarskich od starożytności do współczesności, część I*
Leoński Z., *Węzłowe problemy materialnego prawa administracyjnego, część I-II*
Młyniec E., *Opinia publiczna. Wstęp do teorii*
Mruk H., *Zarządzanie marką*
Nowacka E., *Reformy samorządu terytorialnego w Polsce, Wielkiej Brytanii i USA*
Orliński R., *Rachunkowość finansowa. Zbiór zadań*
Patryas W., *Rozważania o normach prawnych*
Romul J., *Wstęp do teorii państwa i prawa*
Rutkowski T., *Statystyka*
Zimniewicz K., *Podstawy zarządzania*

Dotychczas wydano 18 numerów uczelnianego periodyku „Zeszyty Naukowe WSZiB”, który obecnie jako „Forum Naukowe” ma charakter monograficznych prezentacji efektów prac badawczych pracowników instytucji działających w ramach WSZiB.

Wśród ponad 120 wydanych dotąd i znajdujących się w druku publikacji zwartych przeważają tytuły związane z procesem dydaktycznym (podręczniki, skrypty, przewodniki metodyczne) na wszystkich kierunkach studiów, obok książek stanowiących efekt badań naukowych. Rezultaty badań były również ogłaszane w periodykach naukowych, zbiorach pokonferencyjnych oraz księgach jubileuszowych poświęconych wybitnym profesorom Uczelni (m.in. Profesorom Gerardowi Labudzie, Zbigniewowi Leońskiemu i Wiktorowi Pawlakowi).

Redaktor Naczelny

Prof. zw. dr hab. Włodzimierz Kaczocha

WYŻSZA SZKOŁA ZARZĄDZANIA I BANKOWOŚCI W POZNANIU

WYŻSZA SZKOŁA ZARZĄDZANIA I BANKOWOŚCI W POZNANIU

PLA WE WROCŁAWIU

Działająca na podstawie decyzji MEN z dnia 26.06.1992 r.
Nr DNS 3-TBM/48/92 wraz z późniejszymi zmianami

www.wszib.poznan.pl

SPECJALNOŚCI NA KIERUNKACH

Zarządzanie i marketing

- bankowa obsługa przedsiębiorstw,
- logistyka,
- rachunkowość jednostek gospodarczych,
- reklama i marketing,
- zarządzanie finansami,
- zarządzanie kadrami,
- zarządzanie przedsiębiorstwem,
- zarządzanie sferą usług społecznych,
- informatyka w zarządzaniu,
- przedsiębiorstwo w Unii Europejskiej.

Administracja

- administracja gospodarcza,
- administracja publiczna,
- administracja porządku i bezpieczeństwa publicznego,
- administracja międzynarodowa,
- finanse publiczne,
- polityka administracyjna,
- prawo i ochrona pracy,
- zarządzanie w administracji.

Politologia

- dziennikarstwo i komunikowanie społeczne (z elementami PR),
- stosunki międzynarodowe i europeistyka,
- marketing polityczny.

13-LECIE ISTNIENIA UCZELNI POZNAŃSKIEJ 10-LECIE ISTNIENIA UCZELNI WROCŁAWSKIEJ

TRADYCJA ZOBOWIĄDUJE

- jedno z pierwszych miejsc w Rejestrze MENiS (nr 9/92)
- studia magisterskie (Zarządzanie i Marketing, Administracja) i licencjackie (Politologia)
- Studia eksternistyczne prowadzone za pomocą internetu
- Fundusz Stypendialny im. Jerzego Pietrzyka
- ponad 6 ml PLN z MENiS w 2004 r. dla studentów wszystkich trybów studiów na stypendia socjalne, naukowe i sportowe
- pierwsza w Europie drużyna SIFE (1993)
- pierwsze miejsce pod względem liczby zatrudnionych absolwentów („Newsweek” 2004)
- Europejski Certyfikat Umiejętności Komputerowych (ECDL)
- sześć instytutów naukowych (Marketingu, Samorządu Terytorialnego, Historii Politycznej, Zarządzania, Polityki Społecznej, Podatków i Finansów Publicznych)
- indywidualne granty naukowe
- Dyplom Marszałka Sejmu RP za zasługi w uruchamianiu szkolnictwa niepaństwowego w Polsce
- POZYTYWNA OCENA PREZYDIUM PAŃSTWOWEJ KOMISJI AKREDYTACYJNEJ DOTYCZĄCA JAKOŚCI KSZTAŁCENIA NA KIERUNKU ZARZĄDZANIE I MARKETING (grudzień 2004 r.)

TRADYCJA GWARANTUJE

zamów bezpłatny informator

- 61-538 Poznań, ul. Robocza 4, tel. (061) 835 15 11 • 50-073 Wrocław, ul. Św. Antoniego 24a, tel. (071) 344 49 24 •
- 85-060 Bydgoszcz, ul. Sobieskiego 5, tel. (052) 584 06 92 •



BIBLIOTEKA GŁÓWNA

322345 L/1

Książka prezentuje podstawy informatyki, przygotowując studentów do egzaminu teoretycznego na Europejskie Komputerowe Prawo Jazdy (ECDL), rekomendowane przez Polskie Towarzystwo Informatyczne i uznawane w całej Europie. Dwa pierwsze rozdziały mają charakter ogólnych podstaw, zawierają definicje takich pojęć, jak: informacja i algorytm. Kolejne dotyczą komputera osobistego (elementy składowe konfiguracji sprzętowej i podstawowego oprogramowania komputera oraz komunikacja użytkownika z komputerem), sieci komputerowych i zasad bezpiecznej pracy w sieci. Ostatnie poświęcone zostały bazom danych, które są nieodłącznym elementem wszelkich aplikacji, i wybranym zagadnieniom związanym z zastosowaniami informatyki i jej wpływem na społeczeństwo.

ISBN 83-88544-08-X