

**KRAKOWSKA SZKOŁA WYŻSZA
IM. ANDRZEJA FRYCZA MODRZEWSKIEGO**

MACIEJ PEKALA

**WYBRANE ZAGADNIENIA
INFORMATYKI**

WYDANIE TRZECIE ZMIENIONE I ROZSZERZONE

KRAKÓW 2006

Rada Wydawnicza Krakowskiej Szkoły Wyższej im. Andrzeja Frycza Modrzewskiego:
Klemens Budzowski, Andrzej Kapiszewski, Jacek Majchrowski, Zbigniew Maciąg

Recenzja:

dr Krystyna Moszner

Projekt okładki i stron tytułowych:

Joanna Sroka

Adiustacja:

Karolina Ulman

Korekta:

Kamila Zimnicka-Warchoł



Copyright © by Krakowska Szkoła Wyższa im. Andrzeja Frycza Modrzewskiego, Kraków 2006

ISBN 83-89823-67-5

Żadna część tej publikacji nie może być powielana ani magazynowana w sposób umożliwiający ponowne wykorzystanie, ani też rozpowszechniana w jakiegokolwiek formie za pomocą środków elektronicznych, mechanicznych, kopiujących, nagrywających i innych, bez uprzedniej pisemnej zgody właściciela praw autorskich.

Na zlecenie Krakowskiej Szkoły Wyższej im. Andrzeja Frycza Modrzewskiego

Wydawca: Krakowskie Towarzystwo Edukacyjne sp. z o.o.,

Oficyna Wydawnicza AFM, Kraków, 2006

Sprzedaż prowadzi:

Księgarnia Krakowskiego Towarzystwa Edukacyjnego Spółka z o.o.

Kampus Krakowskiej Szkoły Wyższej im. Andrzeja Frycza Modrzewskiego

ul. Gustawa Herlinga-Grudzińskiego 1, 30-705 Kraków

tel./fax: 012 252 45 93

e-mail: ksiegarnia@kte.pl

Łamanie:

Joanna Sroka

Druk i oprawa:

Tercja s.c.

Wszystkie rysunki zostały opracowane przez autora.

W rysunkach wykorzystano obrazki typu Clipart z pakietu Corel WordPerfect Suite.

SPIS TREŚCI

Od Autora	9
1. Informatyka – co to takiego?	11
2. Czym jesteś, komputerze?.....	13
2.1. Trochę historii	14
2.2. Mikrokomputer od środka	17
3. Sieci komputerowe.....	21
3.1. Budowa i działanie sieci	22
3.2. Zarządzanie siecią	27
3.3. System operacyjny sieci	27
3.4. Zastosowania	28
3.5. Lokalne sieci komputerowe.....	29
3.6. Rozległe sieci komputerowe	29
3.7. Wymiana informacji	30
▼4. Oprogramowanie	31
4.1. Struktura oprogramowania systemów komputerowych	31
4.2. System operacyjny	33
4.3. Dyskowy system operacyjny.....	34
4.4. Komputer osobisty i system MS Windows	35
4.5. Systemy wielozadaniowe i wielodostępne	35
4.6. Języki programowania – kompilatory i interpretery	36
4.7. Porozumiewanie się z komputerem	37
4.7.1. Polecenia	38
4.7.2. Menu	38
4.7.3. Zapytania	38
4.7.4. Okna	39
4.8. Przemysł oprogramowania	39
4.9. Aplikacje	41
4.10. Powstawanie systemów komputerowych i oprogramowania	42
4.11. Etapy rozwoju umiejętności stosowania komputera	43
4.12. Prawo autorskie.....	43
4.13. Bezpieczeństwo i ochrona danych	45

5. Multimedia	48
5.1. Urządzenia techniczne	48
5.2. Cyfrowy zapis obrazu	51
5.3. Grafika komputerowa	52
5.3.1. Grafika rastrowa.....	53
5.3.2. Grafika wektorowa	53
5.3.3. Grafika tekstowa	53
5.3.4. Grafika dwuwymiarowa	54
5.3.5. Grafika trójwymiarowa	54
5.3.6. Grafika nieinterakcyjna	55
5.3.7. Grafika interakcyjna	55
5.3.8. Grafika czasu rzeczywistego.....	55
5.4. Programy do tworzenia obrazu	55
5.5. Programy dla inżynierów	56
5.6. Kompresja danych	56
5.7. Dźwięk w komputerze	57
6. Edycja tekstu.....	59
6.1. O historii redagowania tekstów	60
6.2. Operacje na plikach	60
6.3. Edycja tekstu	61
6.4. Formatowanie na ekranie	62
6.5. Formatowanie wydruku	63
6.6. Znaki na ekranie i na drukarce	63
6.7. Rozszerzenia i udogodnienia	64
6.8. Wymiana tekstów między programami	64
6.9. Skład tekstu – DTP	65
7. Arkusze kalkulacyjne	69
7.1. Trochę historii.....	70
7.2. Zasady działania	70
7.3. Budowanie arkuszy	71
7.4. Arkusz jako baza danych	72
7.5. Wykresy	72
7.6. Modelowanie	76
7.7. Wymiana danych między programami	76
√ 8. Programy prezentacyjne.....	77
8.1. Trochę historii	77
8.2. Zasady prezentacji	78
8.3. Funkcje programu prezentacyjnego	78
8.4. Przygotowanie materiałów.....	79

9. Bazy danych.....	80
9.1. Rys historyczny oprogramowania baz danych	82
9.2. Właściwości bazy danych	83
9.3. System zarządzania bazą danych	84
9.4. Kartotekowe bazy danych	84
9.5. Relacyjne bazy danych	85
9.6. Inne typy baz danych	85
9.7. Modele danych	85
9.8. Budowa bazy danych	86
10. Sztuczna inteligencja.....	87
10.1. Programy operujące na symbolach	88
10.2. Programy wspomaganie decyzji	89
10.3. Rozmowa z komputerem	89
10.4. Automatyczne tłumaczenie	91
10.5. Sieci neuronowe	92
11. Internet i www.....	96
11.1. Trochę historii	96
11.2. Komunikacja w sieci	97
11.3. Adresy i nazwy	97
11.4. Zasoby sieci Internet.....	98
11.5. Dostęp do zasobów sieci Internet	98
11.6. Język HTML.....	100
12. Symulacja	118
Źródła.....	121
Strony www.....	121
Literatura przedmiotu do samodzielnego studiowania	122

OD AUTORA

Współcześnie słowo informatyka używane jest powszechnie, a powoływanie się na wyniki „uzyskane” nie tyle za pomocą komputera, ile przez komputer ma znaczenie niemal magiczne. Ta niewielka książka adresowana jest do osób, które nie zajmują się zawodowo informatyką, ale korzystają z komputera w pracy i w trakcie nauki, a zatem między innymi do studentów kierunków humanistycznych i tych wszystkich, którzy chcą poznać powszechnie używaną terminologię informatyczną.

Zagadnienia omawiane w kolejnych rozdziałach są wycinkiem z szerokiego spektrum tematów zaliczanych do informatyki. Świadomie zostały tu pominięte takie tematy, jak podstawy teoretyczne informatyki, algorytmy i programowanie. Dobór tematów jest odzwierciedleniem cyklu wykładów z przedmiotu „Podstawy informatyki” oraz „Komputerowe metody prezentacji” prowadzonych w Krakowskiej Szkole Wyższej im. Andrzeja Frycza Modrzewskiego.

Cytaty wyróżniono kursywą lub węższą szpaltą, a spis publikacji ich pochodzenia zamieszczono na końcu jako „Źródła”. Definicje podawane na początku rozdziału są również wyróżnione węższą szpaltą.

Na końcu książki zamieszczono wybraną literaturę oraz adresy stron internetowych umożliwiających Czytelnikowi poszerzenie wiedzy. Autor dziękuje kolegom – Dorocie Wilk-Kołodziejczyk, Renacie Urydze i Sebastianowi Rymarczykowi – za pomoc w jej skompletowaniu.

Autor ma nadzieję, że po uważnej lekturze całego opracowania Czytelnik będzie mógł samodzielnie dobrać odpowiednią pozycję w księgarni lub bibliotece, niezależnie od podanej literatury.

1. INFORMATYKA – CO TO TAKIEGO?

Obecnie komputery stały się powszechnym narzędziem w rozwiązywaniu problemów. Co sprawia, że komputer tak wiele potrafi? Otóż tak naprawdę umie on tylko szybko liczyć i dokładnie realizować przekazane mu polecenia. Wszystkie oznaki inteligentnego działania są tylko odzwierciedleniem inteligencji i umiejętności programisty.

Jednak informatyka to nie tylko komputery. Zapoznajmy się najpierw z kilkoma definicjami tego terminu. Każda z nich poprzedzona jest datą jej opublikowania w źródle.

INFORMATYKA:

- [1973] dział cybernetyki stosowanej, zajmujący się budową środków technicznych do przetwarzania informacji i całokształtem zagadnień związanych z ich zastosowaniem. W ujęciu ogólnym przedmiotem badań informatyki są, więc zarówno maszyny matematyczne, w szczególności komputery, jak i nowoczesna dokumentalistyka obejmująca informację naukową, techniczną i ekonomiczną.
- [1975] jest nauką o przetwarzaniu informacji, zwłaszcza przy użyciu automatycznych środków pomocniczych.
- [1976a] zespół dyscyplin naukowych i technicznych zajmujących się przetwarzaniem informacji, zwłaszcza przy użyciu środków automatycznych.
- [1977] nauka o systemach przetwarzania informacji. W ramach informatyki, która zaczyna się kształtować jako odrębna nauka, można wyróżnić następujące kierunki: teoretyczny, konstrukcyjny oraz zastosowaniowy. Pierwsze dwa kierunki informatyki, a mianowicie teoria przetwarzania informacji oraz konstrukcja systemów przetwarzania informacji (sprzęt komputerowy, oprogramowanie) są głównym przedmiotem badań, natomiast zastosowania informatyki z reguły mają znaczenie pomocnicze dla różnych innych specjalizacji.
- [1981] nauka o rozumnym przetwarzaniu, szczególnie przez maszynę automatyczną, informacji traktowanej jako podstawa wiadomości i komunikowania się w dziedzinach technicznych, ekonomicznych i społecznych.
- [1982] dyscyplina naukowo-techniczna zajmująca się całokształtem zagadnień pobierania, przechowywania, przesyłania, przetwarzania i interpretowania informacji.

- [1997] ogół dyscyplin naukowych i technicznych zajmujących się (ogólnie ujmując) informacją, a w szczególności jej komputerowym przetwarzaniem; obejmuje teorie informatyczne, budowanie systemów informatycznych (w tym programowanie), budowę i działanie sprzętu komputerowego, zastosowania metod informatycznych w różnych dziedzinach działalności ludzkiej i in. Teorie informatyczne zajmują się badaniem zjawisk związanych z operowaniem informacją – jej przedstawianiem, przechowywaniem, uzyskiwaniem, porządkowaniem, przetwarzaniem. Prawa rządzące tymi zjawiskami leżą u podstaw budowy narzędzi informatyki będących obiektami fizycznymi (głównie komputerami) i logicznymi (np. algorytmami, językami programowania, programami). Te z kolei stanowią przedmiot badań innych działów teoretycznych informatyki. Teorie informatyki posługują się językiem i metodami matematyki, logiki matematycznej i specyficznych dziedzin, jak teoria języków formalnych i automatów, teoria algorytmów, teoria kolejek; teorie te tworzą również własny język i metody. Programowanie i budowanie systemów informatycznych obejmuje w szczególności tworzenie narzędzi ułatwiających programowanie i ogólnie – posługiwanie się komputerami; do narzędzi tych należą: języki programowania, translatory, systemy operacyjne i inne. Do działu informatyki związanego z programowaniem należą też metody programowania i oceny programów. Dział informatyki związany, ogólnie ujmując, ze sprzętem komputerowym (jego budową i metodami oceny) wykorzystuje zarówno rozwój elektroniki, jak i modele obliczeniowe, tworzone dla szybkiego, algorytmicznego rozwiązywania zadań określonego typu; modele te, oparte np. na zasadzie przetwarzania sekwencyjnego, współbieżnego, potokowego, stanowią teoretyczną podstawę architektur komputerowych, realizowanych następnie elektronicznie. Dzięki stosowaniu metod informatyki wiele różnych dziedzin działalności ludzkiej zostało usprawnionych, np. administracja i zarządzanie, obliczenia numeryczne, sterowanie procesami technologicznymi, przygotowywanie tekstów (edytor), prace wydawnicze (Desktop Publishing), przesyłanie wiadomości (poczta elektroniczna), projektowanie (CAD/CAM), diagnostyka medyczna (np. tomografia komputerowa). Rozwijają się też nowe dziedziny informatyki, takie jak sztuczna inteligencja i grafika komputerowa.

2. CZYM JESTEŚ, KOMPUTERZE?

Tym razem również rozpoczniemy od definicji.

KOMPUTER (elektroniczna maszyna cyfrowa):

- [1973] elektroniczna maszyna cyfrowa, sterowana automatycznie zawartością swej pamięci operacyjnej, w której zapisuje się programy na równi z danymi. Komputer jest więc maszyną matematyczną o działaniu tak zautomatyzowanym, że do posługiwania się nią zbędna jest znajomość wewnętrznych szczegółów konstrukcyjnych, a zadania obliczeniowe wystarczy formułować w odpowiednim języku problemowym.
- [1976a] maszyna matematyczna – zestaw automatycznie działających urządzeń do przetwarzania danych.
- [1977] elektroniczne urządzenie liczące zwane też elektroniczną maszyną cyfrową w celu odróżnienia od mechanicznych urządzeń liczących. Obecnie wyróżnia się czasem także komputery elektroniczne stosowane głównie do obliczeń (tzn. do rozwiązywania problemów naukowych lub technicznych) oraz komputery do przetwarzania danych, stosowane głównie do obliczeń administracyjnych.
- [1981] maszyna cyfrowa – maszyna automatyczna pozwalająca przetwarzać, w ramach programów lub z góry określonych struktur, zbiory operacji arytmetycznych i logicznych do celów naukowych administracyjnych lub księgowości.
- [1997] urządzenie elektroniczne służące do automatycznego przetwarzania informacji (danych) przedstawionych cyfrowo (tzn. za pomocą odpowiednio zakodowanych liczb). Istotną cechą odróżniającą komputer od innych urządzeń jest jego „programowalność”, tzn. wykonywanie konkretnych zadań (np. obliczeń) jest związane z wykonywaniem zapisanych w pamięci komputera programów. Pojęcie komputera obejmuje obecnie zarówno komputery zaprogramowane na stałe, używane jako automaty sterujące, np. w urządzeniach gospodarstwa domowego, jak i komputery uniwersalne, dające się dowolnie zaprogramować.

2.1. TROCHĘ HISTORII

Od najdawniejszych czasów człowiek starał się o to, aby wykonywać obliczenia z jak najmniejszym wysiłkiem, w sposób możliwie automatyczny. Pierwszym, a obecnie często używanym przez dzieci „przyrządem” rachunkowym, były własne palce. Mniej więcej **4000 lat temu** została wynaleziona tabliczka rachunkowa – **abakus (abak)**. Po wielu wiekach udoskonaleń abakus przekształcił się w znane wszystkim **liczydło**.

W roku **1617** szkocki matematyk **John Napier (Neper)** zaprezentował sztabki rachunkowe do obliczania wielokrotności liczb całkowitych.

Już w roku **1642** pojawiło się zupełnie nowe urządzenie liczące. Był nim sumator zaprojektowany przez **Blaise’a Pascala**, słynnego matematyka i filozofa francuskiego. Udoskonolenia sumatora w ciągu ostatnich trzech wieków doprowadziły do powstania czterodziałaniowego arytмомetru, zwanego popularnie „kręciółkiem”. Pierwszy zbudował w latach 1820–1822 francuski konstruktor Xavier Thomas de Colmar, który wymyślił też nazwę *arytмомetr*.

Następnym znaczącym krokiem w kierunku konstrukcji komputera była próba zbudowania przez matematyka angielskiego **Charlesa Babbage’a** **maszyny różnicowej**. W **1822** r. uruchomiono fragment tej maszyny. Babbage stworzył również koncepcję automatycznej i uniwersalnej maszyny liczącej (projekt maszyny „analitycznej”), odpowiadającej w swej strukturze współczesnemu komputerowi z pamięcią (magazynem) i jednostką liczącą (młynem), sterowanej programem zapisanym na kartach dziurkowanych. Takie karty zostały użyte przez J.M. Jacquarda do sterowania krosnem tkackim (1800), a także przez Hermana Holleritha do automatyzacji prac przy spisie powszechnym w Stanach Zjednoczonych (ok. 1890). Tu warto wspomnieć o pierwszej w dziejach kobiecie programistce. Ada Augusta lady Lovelace, córka znanego poety Georga Byrona, zdobywszy wykształcenie matematyczne, co w jej czasach dla kobiet nie było łatwe, opisała pierwsze algorytmy i programy dla maszyny analitycznej.

W 1894 r. Herman Hollerith opatentował system maszyn analityczno-liczących, w których wykorzystał karty dziurkowane. W dwa lata później założył firmę Tabulating Machine, Company (TMC). W 1911 r., łącząc się z innymi firmami, TMC utworzyła, Computing Tabulating Recording (TCR) Company. W 1924 r. Firma TCR przyjęła nazwę International Business Machines, Corporation – IBM, zwaną żargonowo **Big Blue**. Idee Babbage’a z powodu trudności technologicznych musiały czekać na urzeczywistnienie ponad 100 lat. W latach **1939–1944** prof. **Howard M. Aiken** z Uniwersytetu Harvard w powiązaniu z IBM zbudował urządzenie ASCC (ang. Automatic Sequence Controlled Calculator) nazwane **Mark-1**. Wykorzystując postęp techniczny, zastosował elementy elektromechaniczne i zbudował automatyczny kalkulator oparty na pomysłe Babbage’a. Nie był to jednak jeszcze pierwszy komputer.

W latach **1943–1946** na uniwersytecie w Pensylwanii powstał **ENIAC** (ang. Electronic Numerical Integrator And Calculator), pierwsza elektroniczna maszyna licząca, którą zbudowano, bazując na technice lampowej (zawierała 18 tys. lamp elektronowych). Wadą tej maszyny była konieczność wielogodzinnego układania połączeń do wykonania obliczeń i częste awarie lamp elektronowych. Maszyna ta pracowała w dziesiątkowym układzie liczenia. Pomysł **Johna von Neumanna**, aby rozkazy dla maszyny traktować na

równi z liczbami i zastosować dwójkowy układ liczenia, oraz dalszy rozwój techniki doprowadziły do powstania w 1949 r. komputera **EDSAC** (ang. Electronic Delayed Storage Automatic Computer). Został on zbudowany w Anglii na uniwersytecie w Cambridge. Dwa lata później rozpoczął pracę zbudowany w USA analogiczny komputer **EDVAC** (ang. Electronic Discrete Variable Automatic Computer). Te stworzone za pomocą techniki lampowej maszyny zaliczamy dziś do komputerów **pierwszej generacji**.

Postęp techniczny, a w szczególności zastąpienie lamp tranzystorami, zaowocował w 1956 r. komputerami **drugiej generacji**. Prekursorem w tej dziedzinie okazała się firma Bell Laboratories, w której w 1947 r. opracowano tranzystor, a następnie skonstruowano *Leprechaun* – komputer w większości tranzystorowy. Wkrótce tranzystory zostały wyparte przez układy scalone, umożliwiające budowę maszyn **trzeciej generacji**. Był rok 1964. Mniejsze od swych poprzedniczek, były także od nich szybsze i bardziej niezawodne. W tym zakresie prekursorem okazała się firma Texas Instruments, w której w 1958 r. opracowano układ scalony, zastosowany następnie w komputerze IBM 360. Zaledwie pięć lat później, bo w roku 1969 pojawiły się komputery **czwartej generacji**, które zbudowano, bazując na układach scalonych dużej skali integracji (LSI oraz VLSI). Z roku na rok tworzone nowe technologie i typy układów elektronicznych. Oczywiście znajdowało to swoje odbicie w powstawaniu nowych modeli komputerów w ramach czwartej generacji. Jaka zatem będzie piąta generacja komputerów? Aktualnie rysują się cztery drogi:

1. **Komputer białkowy** (biologiczny) oparty na cząsteczkach podobnych do DNA i bakteriach jako elementach pamięci. Pierwszą jaskółką było opracowanie w latach siedemdziesiątych XX w. w IBM diody z organicznych molekuł.
2. **Komputer kwantowy** to układ fizyczny, którego działanie jest bezpośrednio zależne od zjawisk kwantowych. Podstawową jednostką miary w takim komputerze jest kubit (qubit), czyli bit kwantowy. W laboratoriach IBM uruchomiono pierwszy prototyp o mocy obliczeniowej, 7 kubitów.
3. **Komputer optyczny** – w tym zakresie skonstruowano podstawowe układy i uruchomiono pierwsze eksperymentalne egzemplarze optoelektroniczne.
4. **Systemy przetwarzania wiedzy** działają na komputerach zbudowanych w dotychczasowych technologiach, nowością jest sposób rozwiązywania problemów. W latach 1982–1994 w Japonii realizowany interesujący projekt *Knowledge Informations Processing Systems* (KIPS), który miał doprowadzić do powstania komputerów piątej generacji. Mimo wielu interesujących osiągnięć nie uzyskano pełnego sukcesu.

Nową jakością w konstrukcji komputerów stały się w latach siedemdziesiątych XX wieku **mikroprocesory**, zaliczane do układów VLSI. Szybki rozwój technologii doprowadził do wejścia na rynek w drugiej połowie tej dekady **komputerów osobistych** (mikrokomputerów), których cena z roku na rok malała, a możliwości rosły.

Pierwszy mikroprocesor Intel 4004 został skonstruowany w firmie Intel Corporation w 1971 r. na zamówienie japońskiej firmy ETI. Głównym konstruktorem tego mikroprocesora był **Federico Faggin**, późniejszy założyciel firmy Zilog. Rewolucję mikroprocesorową w technice zapoczątkowało wprowadzenie w 1974 r. udoskonalonych mikroprocesorów Intel 8080 i Motorola 68000. W latach 1976–1977 w firmie Zilog

opracowano 8-bitowy procesor Z80. Stał on się podstawą wielu systemów komputerowych, w tym m.in. popularnego w Polsce *ZX Spectrum* firmy Sinclair Research Ltd. W latach 1976–1977 firma Apple Computer Corporation (założyciele Steve Wozniak i Steve Jobs) rozpoczęła seryjną produkcję komputerów Apple II. Natomiast w roku 1983 w sprzedaży pojawił się komputer Apple Lisa z graficznym interfejsem użytkownika. W 1981 r. firma IBM rozpoczęła sprzedaż komputera opartego na mikroprocesorze Intel 8088 o symbolu **PC XT** (skrót od **P**ersonal **C**omputer i **eX**tended **T**echnology). W 1984 r. pojawiły się bazujące na udoskonalonym, 16-bitowym procesorze Intel 80286 komputery oznaczone jako **PC AT** (**A**dvanced **T**echnology). Firma Intel rozwijała swoją rodzinę mikroprocesorów i w 1986 r. wypuściła na rynek 32-bitowy procesor Intel 80386, który w 1988 zastosowano w mikrokomputerze IBM PS/2 model 70. W wyniku rozwoju pojawiły się kolejne procesory 32-bitowe Intel 80486 (1989), a następnie Pentium (1993), Pentium II, Pentium Celeron i Pentium III, Pentium IV. Procesory te są tworzone w architekturze CISC (ang. Complete Instruction Set Computing). Inne firmy, takie jak AMD i Cyrix, też nie zasypiały gruszek w popiele i rozwijały własne, zgodne z kolejnymi produktami firmy Intel, modele procesorów. Firma Motorola rozwijała swoje procesory inną drogą. Są one stosowane w komputerach z rodziny *Macintosh* produkowanych przez firmę Apple od 1984 r. W 1992 r. firma Digital Equipment Corporation wprowadziła na rynek 64-bitowy procesor o nazwie **Alpha**. W połowie lat dziewięćdziesiątych XX wieku firma Apple we współpracy z firmami IBM i Motorola wprowadziła do swoich komputerów 64-bitowe procesory z rodziny **PowerPC**. Procesory te, podobnie jak procesor Alpha, mają architekturę RISC (ang. Reduced Instruction Set Computing/Computer).

Cofnijmy się trochę w czasie, aby naszkicować historię komputerów w Polsce. Pierwszym komputerem był **XYZ** uruchomiony w technice lampowej w 1957 r. W latach 1959–1960 w zakładach Elwro we Wrocławiu opracowano maszynę **Odra 1001**. W roku 1966 przystąpiono do produkcji **Odry 1013**, w rok później pojawiła się **Odra 1204**. Był to już komputer z prawdziwego zdarzenia. Wkrótce wrocławską fabrykę opuściły pierwsze modele z serii **Odra 1300** (**Odra 1304** – 1971, **Odra 1305** – 1972), reprezentujące przyzwoity poziom europejski.

Warto także wspomnieć o skonstruowanym w drugiej połowie lat sześćdziesiątych przez inż. Jacka Karpińskiego **minikomputerze K-202**, wykonującym milion obliczeń na sekundę, a mieszczącym się w walizce. Był to wówczas poziom światowy w tej klasie maszyn.

W drugiej połowie lat osiemdziesiątych pojawiły się polskie konstrukcje mikrokomputerów oparte na procesorze Z80 lub jego odpowiednikach (np. produkowany w NRD U880). Wymienię tu **PSPD-90**, produkowany w Krakowie w zakładach MERA-KFAP, i produkowany we Wrocławiu dla szkół Elwro 800 Junior. Komputery te mogły być łączone w sieć. Pod koniec lat osiemdziesiątych pojawiły się pierwsze składane w Polsce odpowiedniki komputerów PC XT o wdzięcznych nazwach Lidia i Mazovia.

Porównajmy: od abakusa do pierwszego komputera minęło 40 wieków, od pierwszego komputera do komputera osobistego niecałe 35 lat od komputera osobistego do PDA (Personal Digital Assistant) około 15 lat.

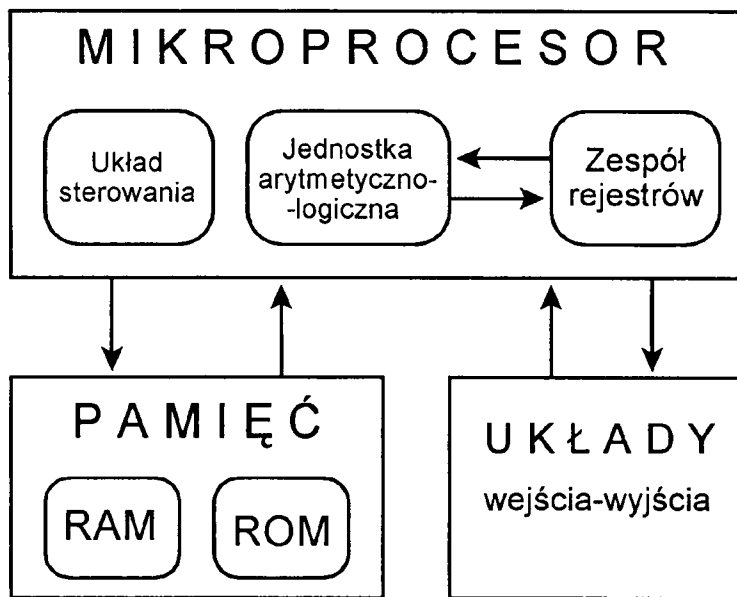
Termin Personal Digital Assistant został ukuty 7 stycznia 1992 przez Johna Sculleya podczas Consumer Electronics Show w Las Vegas (Nevada) w odniesieniu do mikrokomputera Newton firmy Apple.

Obecnie przyjętą się podział komputerów na: superkomputery, duże komputery (*mainframe*), minikomputery i mikrokomputery. Ten podział opiera się przede wszystkim na odmiennych sposobach ich konstruowania oraz biorących się stąd różnicach w ich szybkości i wydajności (mocy obliczeniowej) oraz przeznaczeniu. Superkomputerów używa się głównie do złożonych obliczeń naukowych i technicznych, produkcji grafiki filmowej itp. (np. CRAY), duże komputery przeznaczone są do zastosowań bankowych, finansowych i administracji na szczeblu dużego przedsiębiorstwa (np. IBM 390), minikomputery obsługują małe instytucje lub grupy użytkowników (np. VAX), a obecnie ich miejsce zajęły stacje robocze (np. firmy SUN lub Kayak firmy Hewlett-Packard), mikrokomputery są zwykle przeznaczone dla pojedynczego użytkownika (np. IBM PC, Macintosh).

2.2. MIKROKOMPUTER OD ŚRODKA

Rodowód już znany. Ale jak jest zbudowany komputer?

Omówimy w tym miejscu schemat funkcjonalny mikrokomputera (patrz rys. 2.1.) oraz działanie poszczególnych elementów. Strzałki na rysunku oznaczają przepływ informacji w mikrokomputerze. Nie poprowadzono natomiast strzałek od układu sterowania do pozostałych bloków mikrokomputera, aby nie zaciemniać rysunku.



Rysunek 2.1. Schemat funkcjonalny mikrokomputera.

Sercem mikrokomputera jest **mikroprocesor**. W jego skład wchodzi trzy podstawowe bloki: układ sterowania, jednostka arytmetyczno-logiczna i zespół rejestrów.

Układ sterowania, zgodnie ze swoją nazwą steruje pracą wszystkich bloków mikrokomputera. Jego podstawowym zadaniem jest pobranie zawartości kolejnych komórek pamięci, rozkodowanie i spowodowanie wykonania operacji, odpowiadających rozpoznanym rozkazom. Pracę układu sterowania można wyobrazić sobie, stosując następującą analogię. Dyspozytor w pewnym zakładzie przemysłowym, przychodząc do pracy, zastaje na biurku stos kartek zawierających zadania do realizacji na bieżący dzień. Bierze on kolejne kartki ze stosu i w zależności od ich treści wydaje polecenia, które są realizowane przez zespoły lub pojedynczych pracowników.

Zadaniem **jednostki arytmetyczno-logicznej**, zwanej również **arytmometrem**, jest wykonywanie operacji arytmetycznych i logicznych zgodnie ze wskazaniem układu sterowania. Zatem tutaj, podobnie jak we wspomnianym wyżej przykładzie dyspozytora, poszczególni pracownicy wykonują otrzymane od dyspozytora polecenia.

Zespół rejestrów mikroprocesora jest grupą układów przeznaczonych do przechowywania różnorodnych informacji, które mogą być w danej chwili przydatne. To taki podręczny notes dyspozytora o stanie realizacji bieżących zadań. W skład zespołu rejestrów wchodzi m.in. rejestr rozkazów, licznik rozkazów, akumulator.

Jeden rejestr z tego zespołu, zwany **akumulatorem**, jest wyróżniony, gdyż w nim znajduje się zwykle jeden z argumentów każdej operacji arytmetycznej lub logicznej. Miejsce drugiego argumentu wskazane jest przez układ sterowania na podstawie kodu operacji.

Pamięć, drugi ważny blok mikrokomputera, jest ciągiem układów przeznaczonych do przechowywania informacji w postaci liczb, dokładniej ciągów zer i jedynek. Układy tworzące elementy pamięci nazywamy komórkami. Komórki są ponumerowane, a ich numery nazywamy **adresami**. Jak się zapewne Czytelnik domyślił, stos kartek posiadanych przez dyspozytora obrazuje pamięć mikrokomputera.

Liczby są zapisywane w komórkach pamięci nie w dobrze wszystkim znanym układzie dziesiętkowym, lecz w układzie dwójkowym. Podstawą naszego, pozycyjnego systemu liczenia jest liczba 10. Pamiętamy zapewne ze szkoły: dziesięć jednostek rzędu niższego tworzy jedną jednostkę rzędu wyższego. Zapisując liczby, używamy dziesięciu cyfr: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. W układzie dwójkowym, również pozycyjnym, **dwie jednostki rzędu niższego tworzą jedną jednostkę rzędu wyższego**. Do dyspozycji mamy wówczas tylko dwie cyfry: 0 i 1. W informatyce zamiast mówić **cyfra układu dwójkowego**, używa się zaczerpniętego z teorii informacji terminu **bit**.

W pamięci mikrokomputera bity łączone są w grupy po 8. Grupa taka nosi nazwę **bajt** (*byte*) i oznaczana jest dużą literą B. Zatem 1 KB to 1000 bajtów? Nie, drogi Czytelniku! Uważaj: $1 \text{ KB} = 2^{10} \text{ B} = 1024 \text{ B}$. Tak, 1 KB to 1024 bajty. Tej jednostki używamy, podając wielkość pamięci mikrokomputera. Zatem $640 \text{ KB} = 640 \cdot 1024 \text{ B} = 655\,360 \text{ B}$. Tyle komórek pamięci mieliśmy do dyspozycji w pierwszych komputerach klasy PC XT. Tabela 2.1 zestawia porównanie przedrostków oznaczających mnożniki w komputerach i układzie SI.

Tabela 2.1. Przedrostki i mnożniki

Przedrostek	Komputer	Dziesiętnie	SI
K/kilo-	1 KB = 1024 B = 2^{10} B	1024	$10^3 = 1\ 000$
Mega-	1 MB = 1024 KB = 2^{20} B	1048576	$10^6 = 1\ 000\ 000$
Giga-	1 GB = 1024 MB = 2^{30} B	1073741824	$10^9 = 1\ 000\ 000\ 000$
Tera-	1 TB = 1024 GB = 2^{40} B	1099511627776	$10^{12} = 1\ 000\ 000\ 000\ 000$

Źródło: opracowanie własne.

Często jednak do tej informacji dodaje się trzy duże litery **RAM**. Jest to skrót od angielskiego terminu **Random Access Memory**, to znaczy pamięć o dostępie swobodnym. W tej pamięci zapisywany jest podczas pracy program, dane i niektóre niezbędne komputerowi do pracy informacje, które noszą nazwę zmiennych systemowych. Obszar pamięci zawierający zmienne systemowe jest chroniony przez mikrokomputer przed przypadkowym wpisaniem informacji. Pamięć RAM ma jedną małą wadę: po wyłączeniu zasilania cała zapisana w niej informacja przepada bezpowrotnie. Stąd konieczność przechowywania programu na dysku twardym, dyskietce, CD-ROM-ie lub innym nośniku o trwałym zapisie informacji.

Od tej wady wolna jest druga część pamięci mikrokomputera oznaczona skrótem **ROM**. Pochodzi on od angielskich słów **Read Only Memory**, oznaczających pamięć stałą (dosł. „pamięć tylko do czytania”). Zapisana w tej pamięci informacja nie daje się w prosty sposób wymazać i zastąpić inną. Właśnie w pamięci ROM umieszczony jest zestaw programów niezbędnych do uruchomienia mikrokomputera.

W tym miejscu warto zwrócić uwagę Czytelnika na sposób sterowania pracą układów cyfrowych, w tym mikroprocesora, polegający na dostarczeniu przez zegar sygnału elektrycznego (w założeniu zwykle prostokątnego) o określonej częstotliwości zwanej **częstotliwością taktowania**. Na przykład procesor taktowany częstotliwością 1 GHz wykonuje 1 miliard podstawowych operacji w ciągu sekundy. Jeden rozkaz wymaga od procesora kilku operacji, a to kilka taktów zegara.

Innym elementem charakteryzującym mikroprocesor jest architektura mierzona liczbą bitów. **Architektura k-bitowa** jest to budowa procesora posiadającego wewnętrzne rejestry o długości k bitów, co umożliwia równoczesne przetwarzanie liczby bitów równej k. Architektura k-bitowa **nie** odnosi się do długości szyny danych oraz możliwości adresowych takiego procesora (długości szyny adresowej). Szyna danych to sposób przysyłania danych m.in. między pamięcią a procesorem.

Mamy zatem procesory: 8-bitowe, 16-bitowe, 32-bitowe, 64-bitowe.

Omówimy to szerzej. Procesor 8-bitowy może wykonywać operacje arytmetyczno-logiczne tylko na liczbach ośmiobitowych, to znaczy takich, które dadzą się zapisać w układzie dwójkowym z użyciem najwyżej ośmiu cyfr. Chcąc wykonać operacje na liczbach większych, musimy już korzystać ze specjalnych programów. Drugim ograniczeniem jest to, że

mikroprocesor omawianego typu może przysyłać informacje wyłącznie w grupach ośmiobitowych, czyli informacje jednobajtowe. Aby zwiększyć możliwości mikrokomputera możliwe jest łączenie dwóch bajtów w jedną logiczną (nie fizyczną) całość i wykonywanie na nich pewnych operacji. Wiąże się to przede wszystkim z adresami komórek pamięci i szyną danych. Dysponując adresem jednobajtowym, możemy ponumerować $2^8 = 256$ komórek pamięci, a przy adresie dwubajtowym $2^{16} = 65\,536$, czyli 64 KB. Widać stąd, że dla mikroprocesorów ośmiobitowych pamięć może liczyć co najwyżej 64 KB, licząc razem ROM i RAM. Jeżeli mikrokomputer ma więcej pamięci przy mikroprocesorze ośmiobitowym, to stosowane są dodatkowe układy umożliwiające adresowanie powyżej 65 536. Rozwój procesorów zmierzał w kierunku zwiększenia długości ciągu bitów przetwarzanych równocześnie przez procesor, jak również używanych do adresowania komórek pamięci. Pojawiły się zatem procesory 16-bitowe, a po nich używane obecnie 32-bitowe i 64-bitowe.

Pozostał do omówienia ostatni blok mikrokomputera, a właściwie zespół bloków, czyli **układy wejścia-wyjścia**. Umożliwiają one podłączenie do komputera urządzeń zewnętrznych, dzięki którym przekazujemy polecenia i informacje do komputera oraz otrzymujemy od niego komunikaty i informacje. Układy te umożliwiają podłączenie klawiatury, monitora lub telewizora, dysku twardego, stacji dyskietek (kiedyś magnetofonu kasetowego), drukarki oraz innych urządzeń, na przykład manipulatorów (m.in. myszka, joystick).

W ten sposób omówiony został podstawowy schemat funkcjonalny mikrokomputera. Poszczególne typy mikrokomputerów różnią się między sobą przede wszystkim technicznymi rozwiązaniami wymienionych wyżej bloków oraz ich fizyczną realizacją. A zatem według jakich kryteriów oceniać mikrokomputer?

Pierwszym kryterium jest to, do jakich zadań chcemy go używać. Ale jest to sprawa indywidualna i każdy musi ją rozstrzygnąć we własnym zakresie.

Drugie kryterium to parametry techniczne. Wyróżnić tu trzeba trzy elementy: typ mikroprocesora, wielkość pamięci RAM, możliwość przyłączenia urządzeń zewnętrznych. O możliwościach mikrokomputera decyduje przede wszystkim typ mikroprocesora. Rozpoczęto się od mikroprocesorów ośmiobitowych (na przykład wymienione wyżej), a dziś używamy 32- i 64-bitowych.

Na zakończenie przykładowa konfiguracja komputera na potrzeby domowe w powiązaniu z pojęciami opisanymi w tym rozdziale. Procesor 32-bitowy, np. Intel Pentium IV o częstotliwości zegara 3 GHz (rozwiązanie droższe) lub Intel Celeron 3 GHz (rozwiązanie tańsze), pamięć operacyjna RAM 512MB, pamięci zewnętrzne: dysk twardy 80–160 GB, napęd dyskietek 3,5", nagrywarka, DVD, urządzenia zewnętrzne: klawiatura, manipulator – np. myszka, karta dźwiękowa, karta sieciowa lub faks-modem, monitor kolorowy CRT 17" (rozwiązanie tańsze) lub LCD 17" (rozwiązanie droższe), głośniczki, drukarka atramentowa kolorowa (tańszy zakup, ale droższa eksploatacja) lub drukarka laserowa (droższy zakup, ale tańsza eksploatacja). Fizyczna realizacja komputera, a w szczególności obudowa i możliwości podłączenia innych urządzeń, zależą od producenta i w tym względzie należy skorzystać z pomocy specjalistycznego sklepu lub firmy. Proponowany system operacyjny to MS Windows, aktualnie w wersji XP, choć zapowiadana jest wersja o nazwie Vista.

3. SIECI KOMPUTEROWE

Rozpoczniemy od definicji, a następnie omówione zostaną podstawowe pojęcia.

- [1997] **KOMPUTEROWA SIEĆ** – zespół oddalonych od siebie komputerów i urządzeń, peryferyjnych, połączonych liniami transmisji danych; w przypadku sieci specjalizowanych również urządzeń o specjalnych funkcjach (np. uliczne automaty wydające gotówkę w sieci bankowej, punkty ogniowe w sieciach militarnych i inne). Zależnie od wielkości, rodzaju i stopnia złożoności sieci komputerowej, liniami transmisji danych mogą być zwykłe kable lub linie telekomunikacyjne. Sieć komputerowa może ograniczać się do jednego lub kilku budynków (sieci LAN – ang. Local Area Networks), do miasta, ale może też pokrywać wielkie obszary, kraje, a nawet kontynenty. Ważniejsze cele tworzenia sieci komputerowej: 1) umożliwienie użytkownikom korzystania z odległych zasobów informacyjnych i urządzeń; 2) zwiększenie niezawodności działań dzięki możliwości użycia kilku komputerów do wykonywania tego samego programu, 3) bardziej równomierne wykorzystanie mocy obliczeniowej sprzętu informatycznego (gdy dany komputer jest przeciążony zadaniami, można wykorzystać inny, w danej chwili nieobciążony), 4) usprawnienie zbierania, przetwarzania, rozprowadzania i wykorzystania na rozległych obszarach informacji o specjalnym przeznaczeniu, np. militarnym, gospodarczym, medycznym itp., 5) przesyłanie tekstów (poczta elektroniczna, Internet); 6) upowszechnienie dostępu do informacji z różnych dziedzin, np. nauki, 7) usprawnienie pracy biur podróży, rezerwacji lotniczej, operacji bankowych, handlu, bibliotekarstwa, systemu ubezpieczeń i innych. Sieci mniejsze bywają często podłączane do większych. Taki zhierarchizowany system teleinformatyczny znacznie zwiększa dostępność informacji, zwłaszcza że rozpowszechniło się używanie komputerów osobistych, podłączanych przez modemy do gniazdek telefonicznych w prywatnych mieszkaniach. Szczególnie istotnym elementem przy tworzeniu sieci komputerowej jest ochrona informacji – ułatwiony dostęp do baz danych wiąże się bowiem z niebezpieczeństwem użycia (lub zmiany) pewnych informacji (dotyczących obronności, kont bankowych, sytuacji ekonomicznej przedsiębiorstw i innych) przez

osoby nieuprawnione („włamywaczy” komputerowych – ang. *hacker*). Pierwszą siecią komputerową, uruchomioną jeszcze w latach 50., była amerykańska sieć SAGE (ang. Semi Automatic Ground Environment), służąca do sterowania obroną kontynentu Ameryki Północnej. Obecnie w krajach rozwiniętych istnieje wiele sieci komputerowych, zarówno o charakterze ogólnodostępnym (sieci publiczne abonenckie), jak i o charakterze zamkniętym (sieci wykorzystywane przez konkretne instytucje).

Więcej informacji o historii sieci znajdzie Czytelnik w rozdziale 11.

3.1. BUDOWA I DZIAŁANIE SIECI

Sieć komputerowa składa się z dwóch przenikających się struktur – fizycznej i logicznej, które zostaną omówione w tym podrozdziale.

Informacje możemy przekazać za pomocą sygnałów wytworzonych przez elektryczność, fale radiowe albo światło. Stosuje się dwie techniki przesyłania: analogową – za pomocą fali, oraz cyfrową – w postaci ciągu impulsów reprezentujących poszczególne bity informacji. Transmisja cyfrowa może odbywać się **szeregowo**, tzn. bit po bicie, lub **równoległe** przez wiele kanałów. Do transmisji informacji na duże odległości stosuje się transmisję szeregową.

Do standaryzacji przepływu informacji pomiędzy różnymi urządzeniami w sieci, w szczególności komputerami użytkowników, stosuje się **model OSI** (Open System Interconnection). Model ten został w roku 1984 opracowany przez międzynarodową organizację normalizacyjną ISO (International Organization for Standardization). Dzieli on zadanie przesyłania informacji pomiędzy stacjami sieciowymi na siedem mniejszych zadań zwanych warstwami (patrz tabela 3.1).

Tabela 3.1. Model OSI

Model OSI		
Warstwy wyższe (aplikacje)	Warstwa 7	Aplikacji
	Warstwa 6	Prezentacji
	Warstwa 5	Sesji
Warstwy niższe (transport danych)	Warstwa 4	Transportowa
	Warstwa 3	Sieciowa
	Warstwa 2	Łącza danych
	Warstwa 1	Fizyczna

Źródło: opracowanie własne na podstawie: *Vademecum. Teleinformatyka*, Warszawa 1999, s. 124.

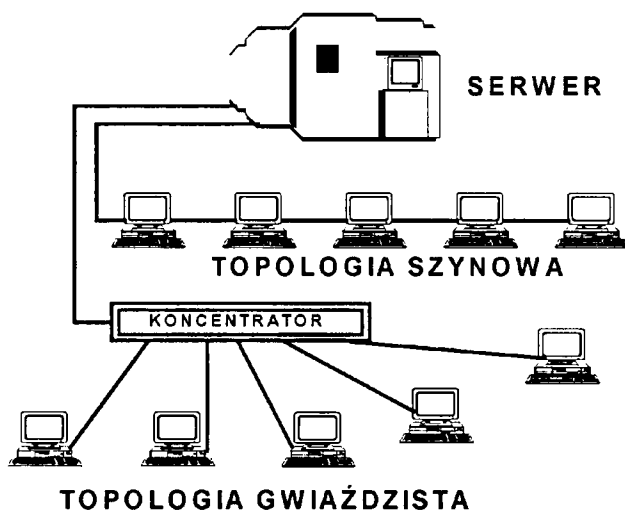
Na strukturze fizycznej można zainstalować różne rodzaje protokołów transmisji. Występują tu pewne standardowe kombinacje. Przedstawię dla przykładu nazwy rodzaju sieci i stosowany rodzaj okablowania.

1. ArcNet – kabel koncentryczny, obecnie nie stosowany, prędkość transmisji 2,5 Mb/s, zasięg do 600 m.
2. Ethernet – kabel koncentryczny gruby albo cienki, obecnie stosowana jest skrętka telefoniczna, prędkość transmisji 10 Mb/s, zasięg do 95 m.
3. FDDI – światłowód, prędkość transmisji ponad 100 Mb/s, zasięg liczony w kilometrach.
4. ISDN – kabel telefoniczny, prędkość transmisji 128 Kb/s, zasięg liczony w kilometrach.
5. xDSL – kabel telefoniczny, prędkość transmisji 16 Kb/s–8 Mb/s, zasięg liczony w kilometrach.

Elementy fizyczne sieci mogą być połączone na kilka różnych sposobów zwanych topologiami. Mamy zatem topologię:

1. szynową (*bus*) – urządzenia połączone są jednym wspólnym kablem jeden za drugim (np. kabel koncentryczny),
2. gwiazdzistą (*star*) – kable sieciowe połączone są w jednym wspólnym punkcie (np. skrętka telefoniczna),
3. pierścieniową (*ring*) – okablowanie tworzy pierścień, do którego podłączone są urządzenia,
4. drzewiastą (*tree*) – zbliżona do szynowej, ale możliwe są gałęzie z wieloma węzłami.

Często stosuje się rozwiązanie mieszane. Na topologię możemy patrzeć dwojako: jako na strukturę fizyczną (okablowanie) oraz jako na sposób logicznej organizacji transmisji.



Rysunek 3.1. Topologia sieci.

Wybór rodzaju sieci zależy na ogół od rozmieszczenia elementów, które mają zostać połączone w sieć w przestrzeni. Inaczej planować musimy sieci w jednym budynku, a inaczej gdy stanowiska rozrzucone są na większym obszarze.

Wprowadzono następującą klasyfikację sieci w tym zakresie:

LAN (Local Area Network) – lokalna sieć komputerowa, na ogół pojęcie to odnosi się do sieci w jednym budynku.

WAN (Wide Area Network) – sieć rozległa, obejmująca swym zasięgiem większy obszar, np. kraj, kontynent, świat.

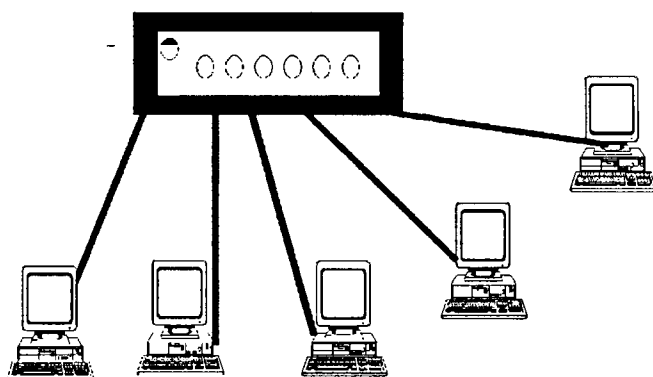
MAN (Metropolitan Area Network) – miejska sieć komputerowa, odnosi się do sieci obejmującej swym zasięgiem miasto. Jest to szczególnie przypadek sieci rozległej.

Mówiąc o sieciach, musimy pamiętać o aktywnych urządzeniach związanych z wysyłaniem i odbiorem sygnału oraz ze wzmocnieniem i rozchodzeniem się sygnału na duże odległości. Chodzi tu m.in. o takie urządzenia, jak karta sieciowa, modem, koncentrator, most (*bridge*) i traser (*router*).

Karta sieciowa (ang. NIC – Network Interface Card) służy do przekształcania pakietów danych w sygnały, które są przesyłane siecią komputerową. Każda karta sieciowa posiada własny, unikalny w skali światowej adres fizyczny, znany jako adres MAC (ang. Media Access Control).

Modem (ang. MOdulator-DEModulator) urządzenie elektroniczne, którego zadaniem jest zamiana danych cyfrowych na analogowe sygnały elektryczne (modulacja) i na odwrót (demodulacja) tak, aby mogły być przesyłane i odbierane poprzez linię telefoniczną (a także łącze telewizji kablowej lub fale radiowe).

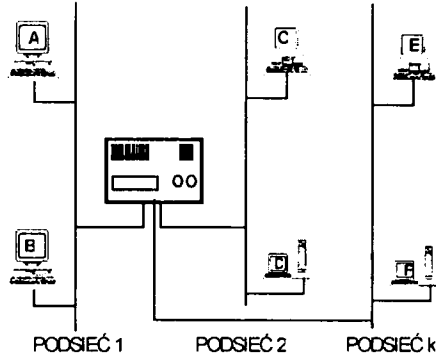
Koncentrator (ang. *hub*) – urządzenie łączące wiele urządzeń sieciowych w sieci komputerowej o topologii gwiazdziej. Jest on związany z fizyczną warstwą sieci (warstwa 1 modelu OSI), (patrz rys. 3.2.).



Rysunek 3.2. Koncentrator.

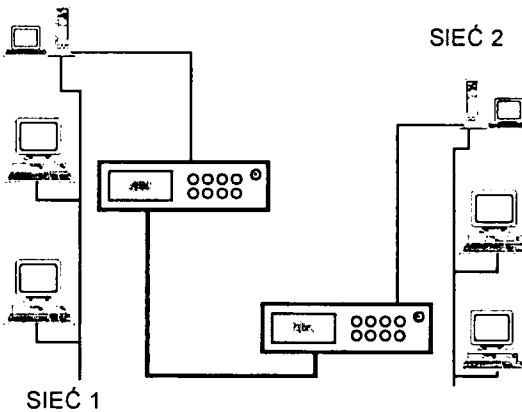
Przełącznik (ang. *switch*) to urządzenie łączące segmenty sieci komputerowej. Przełącznik pracuje w warstwie drugiej modelu OSI (łącza danych), jego zadaniem jest przekazywanie ramek między segmentami sieci. Przełączniki określa się też mianem wieloportowych mostów.

Most (ang. *bridge*) umożliwia separację grup komputerów na podstawie fizycznych adresów. Pozwala to na budowę podsieci w ten sposób, że niektóre komputery nie są widziane przez komputery w innym segmencie sieci. Jest on związany z fizyczną warstwą sieci i protokołem transmisji – warstwa 2 modelu OSI (patrz rys. 3.3.).



Rysunek 3.3. Przełącznik. Most.

Traser (*router*) klasyfikuje i rozdziela pakiety na podstawie adresu docelowego i dokonuje wyboru drogi, którą ten pakiet będzie dążył do celu. Służą temu zdefiniowane przy instalacji trasera tablice. Jest on związany z fizyczną, sieciową i transportową warstwą sieci – warstwa 3 modelu OSI (patrz rys. 3.4.).



Rysunek 3.4. Trasery.

Koncentratory (huby) znajdują zastosowanie głównie w sieciach LAN, natomiast przełączniki (swicche) i trasery (routery) w sieciach MAN i WAN. Szczególnie sieci WAN nie mogą istnieć bez traserów.

Spośród wielu usług i protokołów sieci wymienię kilka, które spotyka się najczęściej. Przez protokół sieci rozumie się ustalony standard danych i sposób ich przesyłania.

EDI (Electronic Data Interchange) – podstawą systemu elektronicznej wymiany danych jest przesyłanie komunikatów elektronicznych (dokumentów) między różnymi i odległymi systemami komputerowymi. Do tego celu stosuje się uzgodniony sposób przedstawienia danych UNI EDIFACT (Unique Electronic Data Interchange for Administration, Commerce and Transport), obowiązujący w Polsce jako PN-90/T-20091. Używa się również kodów kreskowych systemu EAN, poczty elektronicznej X.400 i adresacji usług katalogowych X.500. Systemy komputerowe nadawcy i odbiorcy komunikatów EDI mogą być całkowicie odmienne, a jedynym warunkiem jest właściwa interpretacja standardowej struktury danych. Głównymi obszarami zastosowań EDI są aplikacje gospodarcze klasy e-biznes, a w szczególności handel elektroniczny *e-commerce* jako jedno z zastosowań aplikacji e-biznes w sieci Internet.

SMTP (Simple Mail Transfer Protocol) – stosowany w Internecie protokół pocztowy umożliwiający wymianę poczty między serwerami pocztowymi oraz wysyłanie poczty przez użytkownika..

POP3 (Post Office Protocol, version 3) – protokół poczty elektronicznej stosowany w Internecie do odbierania poczty z serwera.

PPP (Point-to-Point Protocol) – jeden z dwóch standardowych protokołów umożliwiających połączenie komputera z Internetem przez zwykłe łącze telefoniczne. Wyparł starszy protokół SLIP.

SNMP (Simple Network Management Protocol) – najczęściej obecnie stosowany protokół do zarządzania sieciami komputerowymi.

ATM (Asynchronous Transfer Mode) – sposób transmisji danych w sieci komputerowej, w którym przesyłane dane dzieli się na fragmenty po 53 bajty, zwane komórkami, oraz nawiązuje się połączenie między nadawcą a odbiorcą. Dzięki takiemu rozwiązaniu możliwe jest jednoczesne i bezkolizyjne przesyłanie głosu, dźwięku i danych komputerowych tą samą linią.

TCP/IP (Transmission Control Protocol/Internet Protocol) – termin ten używany jest obecnie jako skrót dla stosowanego w Internecie zestawu protokołów (ok. 100 współpracujących). Więcej na ten temat w rozdziale 11.

ISDN (Integrated Services Digital Network) – jest cyfrowym systemem telekomunikacyjnym z integracją usług. Zapewnia on usługi oferowane dotychczas w sieciach analogowych oraz przekazy cyfrowe oparte na technice cyfrowego przetwarzania informacji, komutacji kanałów i pakietów. Sieci ISDN umożliwiają tworzenie lokalnych centralek domowych, do ośmiu urządzeń, a ich możliwości przewyższają aktualne potrzeby użytkowników. Aby skorzystać z usług sieci cyfrowej, należy stosować terminale (w tym telefony cyfrowe) oraz uaktywnić odpowiednie usługi telekomunikacyjne. Dostęp mamy przez używaną do tej pory dwuprzewodową linię telefoniczną między abonentem a centralą. Posiadane telefony analogowe można przyłączyć do sieci ISDN przez odpowiednie urządzenie, ale będą one funkcjonować w zakresie podstawowym.

xDSL (Digital Subscriber Line – cyfrowa linia abonencka) jest technologią szerokopasmowego dostępu do Internetu. Technologia stosowana jest w mieszkaniach prywatnych i małych firmach, a dzięki odpowiednim filtrom dostępne są używane do tej pory standardowe usługi telefoniczne, przede wszystkim głosowe, bez konieczności wymiany używanych do tej pory urządzeń telefonicznych.

Litera x na początku skrótu oznacza, że jest to cała rodzina rozwiązań technologicznych opartych na wspólnym trzonie. W szczególności:

IDSL (ISDN Digital Subscriber Line, wariant ISDN)

ADSL (Asymmetric Digital Subscriber Line)

HDSL (High Bit Rate Digital Subscriber Line)

RDSL (Rate Adaptive Digital Subscriber Line)

SDSL (Symmetric Digital Subscriber Line)

VDSL (Very High Speed Digital Subscriber Line)

3.2. ZARZĄDZANIE SIECIĄ

Ważnym elementem utrzymania sprawności sieci jest możliwość zarządzania nią. Omówiony zostanie pokrótce model zarządzania siecią według wspomnianej wcześniej organizacji ISO. Model składa się z pięciu głównych elementów:

1. **Zarządzanie wydajnością** – celem jest utrzymanie wydajności na odpowiednim poziomie poprzez pomiar, stałą kontrolę i analizę podstawowych parametrów sieci.
2. **Zarządzanie konfiguracją** – jego celem jest zapewnienie współdziałania różnych wersji sprzętu i oprogramowania poprzez monitorowanie sieci i informacji konfiguracyjnych.
3. **Zarządzanie sprawozdawczością**, które ma regulować wykorzystanie sieci przez pojedynczego użytkownika lub ich grupę. Realizuje się to, mierząc wykorzystanie zasobów sieciowych przez poszczególnych użytkowników, a wyniki pomiarów mogą być wykorzystane do ustalenia odpłatności za pracę w sieci.
4. **Zarządzanie uszkodzeniami**, którego celem jest wczesne wykrycie uszkodzeń sieci i powiadamianie użytkowników o ich wystąpieniu, przy równoczesnym podjęciu prac naprawczych.
5. **Zarządzanie bezpieczeństwem** ma zapewnić dostęp do zasobów sieci w sposób autoryzowany. Podsystem zarządzania bezpieczeństwem umożliwi dostęp autoryzowanym użytkownikom do zasobów sieci oraz monitorowanie dostępu do szczególnie istotnych zasobów sieciowych.

3.3. SYSTEM OPERACYJNY SIECI

Ważnym elementem jest system operacyjny sieci, na przykład – Novell NetWare i Unix. Każdy ma inne właściwości i jego wybór musi być starannie przemyślany. Podam kilka uwag związanych z tymi systemami.

Novell NetWare jest bardzo dobrym systemem operacyjnym dla sieci budowanych z komputerów już pracujących w przedsiębiorstwie, na których eksploatowane są systemy związane z działalnością firmy (np. w wykorzystywanym na komputerze użytkownika systemie MS DOS). Możliwa jest dalsza eksploatacja tych programów w sieci w dotychczasowym trybie.

System **Unix** jest systemem wielozadaniowym. Nie umniejszając jego roli dla sieci LAN, trzeba podkreślić jego użyteczność w sieciach MAN i WAN. Centralne komputery tych

sieci pracują właśnie w systemie operacyjnym Unix. Przykłady systemów z tej grupy to: **System V** (AT&T), **BSD UNIX** (Uniwersytet w Berkeley), **OSF/1** (Open Software Foundation), **Solaris** (SUN), **HP UX**, (Hewlett-Packard), **IBM AIX**, **XENIX** (dla komputerów PC), **A/UX** (dla komputerów Macintosh), **Linux**.

Różnica między systemem NetWare i Unix przejawia się między innymi w tym, że w sieci Unix wszystkie programy realizowane są na serwerze sieci, a w NetWare na komputerze, z którego w danej chwili korzysta użytkownik.

Koszty pierwszej instalacji i rozbudowy sieci są różne w obu systemach. Ostateczna decyzja zakupu takiego lub innego systemu może być podjęta po dokładnej analizie potrzeb i kosztów. Wymaga to zarówno konsultacji, jak i współpracy ze specjalistami.

3.4. ZASTOSOWANIA

Techniki klient-serwer

Klient-serwer jest asymetrycznym rozwiązaniem stosowanym w oprogramowaniu, w którym rozdziela się pewne funkcjonalności. Umożliwia to zwiększenie elastyczności i ułatwia wprowadzanie zmian w każdej części. Z serwera korzysta na ogół wielu klientów, a w przypadkach szczególnych jeden klient może korzystać z kilku serwerów.

Klient to oprogramowanie, które korzysta z usług dostarczanych przez serwer. Dla konkretnego serwera wymagany jest odpowiedni klient. Najpopularniejszym układem klient-serwer są przeglądarki internetowe (klient) i serwer stron internetowych np. Apache. Inny przykład to oprogramowanie grupy telnet lub SSH, np. PuTTY, które umożliwia zdalny dostęp do zasobów komputera pracującego w systemie operacyjnym klasy UNIX. Technika ta jest także wykorzystywana w systemach do zarządzania lub innych systemach, w których wymagany jest dostęp do bazy danych.

Dla zamknięcia klasyfikacji należy wspomnieć o produkcie MS Windows 2003 server firmy Microsoft. Zyskuje on coraz większą popularność, mieszcząc się ze swoją funkcjonalnością pomiędzy Novell Netware a Unixem.

P2P

P2P to skrót angielskiego terminu *peer-to-peer* (równy z równym) oznaczającego sposób komunikacji pomiędzy komputerami w sieci. Ten model gwarantuje obydwu stronom równorzędne prawa, natomiast w modelu klient-serwer, serwer ma uprzywilejowaną pozycję. W sieci P2P każdy z komputerów jest zarazem klientem i serwerem. Wymiana danych prowadzona jest bez pośrednictwa centralnego serwera i odbywa się bezpośrednio między dwoma komputerami w sieci. Sieć P2P ma zmienną strukturę zależną od komputerów, które są w niej aktualnie zalogowane.

Najczęściej spotykanym zastosowaniem sieci P2P w Internecie są programy do wymiany plików. Mogą to być między innymi pliki tekstowe, muzyczne, filmy, programy. Pierwszym rozwiązaniem stosowanym na masową skalę od 1999 r. był Napster założony przez Shawna Fanninga. Popularność swą zawdzięczał przede wszystkim łatwości obsługi. Obecnie dużą popularnością cieszą się np. KaZaA, e-Mule, Gnutella. Rozwiązanie P2P budzi wiele kontrowersji ze względu na naruszanie praw autorskich.

Często udostępnia się pliki, które nie występują na oficjalnych serwerach ze względu na brak zgody właściciela praw autorskich, głównie majątkowych. Udostępnia się jednak wiele plików, które nie są chronione prawami autorskimi lub właściciele praw autorskich wyrazili zgodę na ich udostępnianie tą drogą. Sieci P2P są też wykorzystywane do omińnięcia cenzury lub innych ograniczeń nałożonych przez prawo.

W Polsce samo pobieranie plików, szczególnie muzycznych, nie jest nielegalne zgodnie z art. 23 Ustawy o prawie autorskim i prawach pokrewnych dotyczącym dozwolonego użytku osobistego. Naruszeniem prawa jest dopiero rozpowszechnianie uzyskanych tą drogą utworów.

Praca grupowa – współdzielenie drukarek, programów i plików.

Sieć komputerowa umożliwi efektywniejsze wykorzystanie urządzeń w firmie, w szczególności drukarek, skanerów i faksów. Laserowe drukarki sieciowe, zarówno kolorowe, jak i czarno-białe, są urządzeniami wysokowydajnymi o niskich kosztach eksploatacji.

Udostępnianie plików, w szczególności dokumentów, umożliwia ograniczenie ilości zużywanego papieru oraz łatwy dostęp do postaci elektronicznej wzorów dokumentów firmowych. W rozwiniętej postaci prowadzi to do rozwoju biura elektronicznego i pracy grupowej.

3.5. LOKALNE SIECI KOMPUTEROWE

Już dwa komputery klasy PC z systemem operacyjnym MS Windows XP i połączone odpowiednim kablem tworzą lokalną sieć komputerową. Komputery wyposażone w karty sieciowe można za pomocą odpowiednich kabli podłączyć do koncentratora, tworząc w łatwy sposób sieć w jednym pomieszczeniu. To kolejny przykład lokalnej sieci komputerowej.

Zbudowanie sieci lokalnej, obejmującej swym zasięgiem cały budynek, jest złożonym przedsięwzięciem, które rozpoczyna się od planowania topologii sieci, rozmieszczenia urządzeń aktywnych i serwera (lub serwerów), a kończy podłączeniem komputerów w poszczególnych pomieszczeniach na biurkach pracowników. Taką sieć najczęściej posiadają użytkownicy komputerów w swoim miejscu pracy.

3.6. ROZLEGŁE SIECI KOMPUTEROWE

Współczesne sieci komputerowe obejmują swym zasięgiem cały świat i pozwalają ich użytkownikom korzystać z różnorodnych usług. Między innymi umożliwiają dostęp do światowych baz danych z różnorodnych dziedzin. Jednak najczęściej wykonywaną usługą jest przesyłanie informacji pomiędzy użytkownikami. Oprócz prostej wymiany może ona być realizowana przez listy dystrybucyjne, grupy dyskusyjne, serwisy i systemy informacyjne. Usługa ta znana jest jako poczta elektroniczna (e-mail). Tego rodzaju poczta służy nie tylko do przesyłania tekstów, ale również programów, baz danych i plików multimedialnych.

Rozległe sieci komputerowe mogą być używane do kierowania przedsiębiorstwami o dużym zasięgu terytorialnym.

3.7. WYMIANA INFORMACJI

W odniesieniu do przytoczonej na początku rozdziału definicji sieci komputerowej przyjrzyjmy się teraz niektórym ogólnie spotykanym terminom określającym sieci komputerowe.

Internet (dosł. międzysieć; ang. *inter* – między, *net* – sieć) jest siecią komputerową o światowym zasięgu łączącą sieci lokalne, sieci rozległe i wszystkie komputery do nich podłączone.

Intranet to sieć komputerowa przedsiębiorstwa (firmy, organizacji) wyposażona w serwer umożliwiający korzystanie z takich usług jak strony WWW, poczta elektroniczna itp., czyli usług typowo internetowych. Do intranetu dostęp mają zazwyczaj tylko pracownicy danego przedsiębiorstwa.

Ekstranet jest to połączenie co najmniej dwóch intranetów za pomocą protokołów sieciowych. Ekstranety tworzy się w celu wzajemnego udostępnienia zasobów między przedsiębiorstwami (organizacjami) lub między nimi i klientami, przy równoczesnym zabronieniu powszechnego dostępu z sieci Internet.

Teraz zwróćmy uwagę na WWW, gdyż skrót ten nie oznacza sieci, lecz usługi w sieci.

World Wide Web (w skrócie WWW lub Web) jest hipertekstowym, multimedialnym, sieciowym systemem informacyjnym opartym na publicznie dostępnych, otwartych standardach IETF, W3C i ISO. Pierwotnym i w chwili obecnej nadal podstawowym zadaniem WWW jest publikowanie informacji. Jest on właściwą częścią Internetu i nie jest z nim tożsamy.

4. OPROGRAMOWANIE

Oprogramowanie to całość programów napisanych z uwzględnieniem cech sprzętu komputerowego umożliwiających jego wykorzystanie. Programy, wykonując ciągi instrukcji, umożliwiają przetwarzanie danych w określonym przez twórcę programu zakresie. W skład oprogramowania wchodzi zarówno programy komputerowe, jak i aplikacje. Oprogramowaniem zajmuje się jeden z działów informatyki.

4.1. STRUKTURA OPROGRAMOWANIA SYSTEMÓW KOMPUTEROWYCH

Od zarania oprogramowanie maszyn cyfrowych dzielono na dwie grupy – oprogramowanie systemowe (zwane też podstawowym) oraz użytkowe.

Pierwszą grupę tworzą systemy operacyjne, kompilatory języków programowania, programy i podprogramy związane z poszczególnymi kompilatorami oraz programy pomocnicze.

System operacyjny jest zestawem programów umożliwiających komunikację użytkownika z komputerem oraz realizację jego programów i aplikacji. System operacyjny pośredniczy w wymianie informacji między aplikacjami a sprzętem.

Systemy operacyjne można podzielić według następujących cech:

- **jednoprogramowość**, gdy programy realizowane są szeregowo, jeden po drugim,
- **wieloprogramowość**, gdy programy realizowane są równocześnie.

Spotykamy dwa tryby pracy systemu komputerowego – przetwarzanie wsadowe i zdalne przetwarzanie danych (por. „Wielodostęp” w podrozdz. 4.5). Nazwa przetwarzanie wsadowe wywodzi się z czasów, gdy programy i dane wczytywało się do systemu komputerowego z nośników papierowych, takich jak karty i tasiemki.

- **wirtualizacja pamięci**, wszystkie obszary pamięci systemu są traktowane jednorodnie, bez względu na to, czy są one związane z pamięcią operacyjną komputera, czy też z urządzeniami pamięci zewnętrznymi,
- **rozdzielność architektury** procesora i systemu wyraża się przekazaniem znacznej części dotychczasowych funkcji systemu operacyjnego wyspecjalizowanym urządzeniom. Tak zbudowane systemy komputerowe, o strukturze modułowej, są lepiej przystosowane do zdalnego przetwarzania danych i są przez nie sterowane.

Wśród programów pomocniczych można wyróżnić trzy grupy:

- obsługi technicznej,
- obsługi procesu przetwarzania,
- przygotowania zadań dla systemu.

Programy obsługi technicznej służą zazwyczaj do sprawdzania sprzętu i lokalizacji uszkodzeń zarówno w całym systemie, jak i w jego elementach składowych. Programy diagnostyczne można podzielić na dwa typy – włączone do systemu operacyjnego i tworzące z nim integralną całość oraz wprowadzane do systemu i uruchamiane zgodnie z zaistniałymi potrzebami. Dla komputerów klasy PC z systemem operacyjnym MS Windows są to np. programy o nazwie *scandisk* i *defrag* oraz zewnętrzny pakiet oprogramowania *Norton Utilities*.

Do **programów obsługi procesu przetwarzania** zaliczamy przede wszystkim programy kopiujące i ładujące. Programy kopiujące pozwalają nam przenosić zawartości określonych obszarów danych w systemie pomiędzy różnymi urządzeniami systemu. Programy ładujące na ogół wykorzystuje się do wprowadzenia do pamięci systemu programów lub danych, gdy nie można wykorzystać do tego systemu operacyjnego, np. podczas lub po awarii. Czy korzystałeś kiedyś, Czytelniku, z kombinacji klawiszy Shift+PrtScr, a następnie w programie MS Paint z kombinacji Ctrl+V, aby uzyskać kopię obrazu na ekranie? To przykład przeniesienia danych między różnymi urządzeniami w komputerze klasy PC z systemem operacyjnym MS Windows.

Programy używane w procesie przygotowania zadań dla systemu to głównie programy wykonujące aktualizację, sortowanie i łączenie plików, a także programy wyszukiwania informacji, opracowywania i wyprowadzania sprawozdań. Podobnie jak programy obsługi technicznej, na ogół są dostarczane wraz z systemem operacyjnym komputera i uzupełniane podczas eksploatacji. W systemie MS Windows korzystałeś zapewne, Czytelniku, z umieszczonej w menu „Start” opcji wyszukiwania plików.

Ogromną rodzinę tworzy oprogramowanie użytkowe. Są to programy, które umożliwiają użytkownikowi rozwiązywanie praktycznych problemów technicznych, gospodarczych, naukowych i innych. Źródłem programów może być: producent sprzętu komputerowego, wyspecjalizowana firma produkująca oprogramowanie, inni użytkownicy tego samego sprzętu, a także sam użytkownik.

Wśród oprogramowania użytkowego znajdziemy edytory tekstu oraz DTP, arkusze kalkulacyjne, bazy danych, przeglądarki internetowe, programy do zarządzania, w tym finansowo-księgowo, oraz związane z rozrywką, np. gry.

Omówiony powyżej podział jest umowny i często nie można jednoznacznie określić przynależności danego programu do którejś z wymienionych grup. Jakby nie dzielić oprogramowania systemów komputerowych, postuluje się, aby miało ono określone cechy. Wyróżniamy tu m.in. **przenośność**, czyli korzystanie z programów w różnych systemach komputerowych. Cecha ta jest pożądana dla oprogramowania użytkowego, które powinno pracować w wielu środowiskach. Na ogół nie dotyczy ona systemów operacyjnych. Jako synonimu przenośności używa się terminu **niezależności oprogramowania od maszyny**.

Przystosowalność oznacza zdolność dostosowania programu do potrzeb użytkownika bez konieczności zmiany otoczenia. Należy zwrócić uwagę, że przenośność dotyczy możliwości zmian otoczenia programu, a przystosowalność – algorytmu. Czasami, dla podkreślenia możliwości wprowadzania zmian do programu, używa się pojęcia **elastyczności** programu.

Ważną obecnie cechą oprogramowania systemów cyfrowych jest ich **rozszerzalność (skalowalność)**. Cecha ta wskazuje na możliwości rozbudowy systemu komputerowego i wykorzystywanego oprogramowania w miarę jego eksploatacji. Nadanie oprogramowaniu struktury modularnej umożliwia osiągnięcie znacznej rozszerzalności oprogramowania. Wówczas moduł (element) oprogramowania musi mieć ściśle zdefiniowane zadania oraz sposób łączenia z innymi modułami. Dzięki temu łatwiejsze jest budowanie systemów złożonych oraz ich dokumentowanie i konserwacja.

Przenikalność wiąże się z możliwością udostępnienia użytkownikowi elementów architektury logicznej systemu cyfrowego w wykorzystywanym oprogramowaniu. Cecha ta obecnie bardzo straciła na znaczeniu, gdyż nawet producenci oprogramowania korzystają z bibliotek dostarczonych przez producenta systemu operacyjnego.

Uniwersalność wyżej wymienionych cech oprogramowania systemów komputerowych umożliwia ich ogólną klasyfikację i charakterystykę.

4.2. SYSTEM OPERACYJNY

Komputer jest systemem złożonym z wielu połączonych ze sobą i współdziałających elementów. Jednocześnie przebiega w nim wiele różnych procesów. Z tego powodu potrzebne jest oprogramowanie, które zarządza wszystkimi elementami systemu i organizuje jego pracę w możliwie najbardziej efektywny sposób. To oprogramowanie, będące zestawem programów, nazywamy systemem operacyjnym. Po włączeniu komputera podstawowe procedury systemu operacyjnego są wprowadzane z pamięci zewnętrznych do pamięci operacyjnej komputera. Im bardziej złożony jest komputer, tym bardziej rozbudowany i skomplikowany jest jego system operacyjny.

System operacyjny realizuje między innymi następujące podstawowe funkcje:

- zarządza zasobami komputera, m.in. procesorem (a dokładniej czasem procesora), pamięcią, urządzeniami peryferyjnymi oraz przydziela zasoby procesom,
- koordynuje pracę wyżej wymienionych urządzeń poprzez obsługę przerw i odpowiednio na nie reagowanie,
- chroni dane i pamięć – tak aby jeden proces w wyniku błędu lub zamierzonego działania nie mógł zniszczyć lub pozyskać danych innego procesu,
- automatyzuje najczęściej wykonywane funkcje,
- ukrywa skomplikowanie sprzętu komputerowego przez tworzenie abstrakcji, np. zbiory zapisanych klastrów na dysku widziane są jako pliki o symbolicznych nazwach, programy wykonywane na przemian, wydają się wykonywane równocześnie,
- zapewnia jednolity sposób dostępu do urządzeń,
- sprawdza stan klawiatury i reaguje odpowiednio na naciśnięcie klawiszy,

- wykonuje polecenia użytkownika umożliwiające zarządzanie programami: ładowanie ich do pamięci i zapamiętywanie na dysku, uruchamianie, zatrzymywanie i usuwanie z pamięci,
- informuje o błędach w pracy systemu, programów i urządzeń zewnętrznych.

Ogólnie przyjmuje się, że system operacyjny składa się z trzech głównych elementów:

- jądra systemu wykonującego przede wszystkim zadania wymienione powyżej,
- powłoki, czyli programu realizującego komunikację użytkownika z systemem operacyjnym,
- systemu plików, czyli struktury zapisu danych, przede wszystkim na dysku.

Komunikacja pomiędzy poziomem maszynowym a poziomem systemu operacyjnego odbywa się poprzez interpreter poleceń, którym posługują się programy systemowe do komunikacji z jądrem systemu operacyjnego, natomiast samo jądro systemu działa już bezpośrednio na sprzęcie komputerowym.

Przykładowe systemy operacyjne komputerów to:

firma Apple: Mac OS, Mac OS X, A/UX,

firma IBM: PC DOS, OS/2, AIX, OS/400, OS/390, VM/CMS, DOS/VSE, OS/360, MFT, MVT, SVS, MVS, TPF, ALCS, z/OS,

firma Microsoft: MS-DOS, Microsoft Windows 1.0, 2.0, 3.x, 95/98/Me, CE, NT, 2000, XP, Server 2003.

4.3. DYSKOWY SYSTEM OPERACYJNY

System operacyjny komputerów wyposażonych w pamięci dyskowe musi zapewnić obsługę tych urządzeń oraz gospodarowanie plikami zapisanymi na dysku. Współczesne systemy operacyjne w różny sposób organizują strukturę fizyczną, natomiast w podobny sposób przedstawiają strukturę logiczną, grupując pliki w katalogi.

Jeden z najbardziej znanych i najstarszych nosi nazwę DOS (skrót od ang. **D**isc **O**perating **S**ystem – dyskowy system operacyjny). Zapisany był na ogół na dysku twardym, ewentualnie na dyskietce. Obecnie został zastąpiony przez MS Windows lub Linux. W popularnym komputerze klasy IBM PC podczas startu komputera część systemu operacyjnego jest przepisywana do pamięci RAM. Ta rezydentna część realizuje najczęściej wykonywane operacje. Inne elementy systemu operacyjnego są ładowane w razie potrzeby i usuwane po zakończeniu swego działania.

W systemie DOS zarówno dysk, jak i dyskietka mają specjalną organizację. Podzielone są na ścieżki i sektory. Z tego powodu dane transmitowane są blokami o ustalonej długości odpowiadającej sektorom. W nowszych systemach można od razu przesłać kilka sektorów, a to przyspiesza pracę komputera. W pamięci RAM jest zarezerwowane miejsce, zwane buforem, w którym umieszcza się odczytane lub przygotowane do zapisu dane.

Nowocześniejsze systemy operacyjne organizują inaczej strukturę dysków, dzięki czemu efektywniej gospodarują miejscem oraz umożliwiają obsługę dysków o większych pojemnościach. Podstawowym elementem jest nie sektor, lecz klaster. Tak jest np. w używanym w MS Windows XP systemie plików NTFS (New Technology File System).

Programy lub zbiory danych są w pamięci dyskowej zorganizowane od strony logicznej w pliki (ang. *file*).

System operacyjny oprócz plików danych i programów zapisuje dodatkowe informacje używane przez system operacyjny – katalog plików, inaczej folder (ang. *directory*), czyli informacje o tym, gdzie na dysku znajdują się elementy pliku. Podstawowe operacje dyskowe można podzielić na dwie grupy. Pierwsza zawiera operacje zapisu i odczytu danych oraz sterowanie pracą dysku. Tym zajmuje się system operacyjny i użytkownik nie ma do tych danych dostępu. Do drugiej grupy można zaliczyć operacje związane z zarządzaniem plikami:

- tworzenie plików, czyli przydzielanie im miejsca na dysku, zapisywanie wszystkich rekordów i nadawanie nazwy,
- odczytywanie zawartości plików,
- zmienianie nazw plików,
- usuwanie plików – ta operacja zwalnia miejsce na dysku,
- kopiowanie plików oraz całych ich struktur, np. dyskietek,
- ochronę dostępu do danych, np. blokadę zapisu na dyskietce,
- tworzenie, pamiętanie i drukowanie katalogu plików, czyli pliku o specjalnej strukturze.

Ponadto system zawiera programy wykonujące operacje formatowania dyskietek i dysków, czyli wstępnego ich przygotowania do pracy, a w szczególności podziału na sektory lub klastry, utworzenia katalogu itp.

4.4. KOMPUTER OSOBISTY I SYSTEM MS WINDOWS

Jak wskazano wyżej, oprogramowanie komputera możemy podzielić na dwie grupy – systemowe i użytkowe. Oprogramowanie systemowe zarządza całym komputerem, umożliwiając także pracę programów użytkowych. Te ostatnie realizują zadania użytkownika.

Podstawowe programy systemowe w komputerze osobistym to BIOS (Basic Input/Output System) i system operacyjny. Obecnie najpopularniejszym systemem osobistym komputera klasy PC jest produkt firmy Microsoft zwany Windows w wersjach 95, 98, Millenium, 2000 oraz XP. MS Windows w wersjach 95 i 98 zawiera w sobie dyskowy system operacyjny MS DOS, czyli swego poprzednika. MS Windows, niezależnie od wersji, realizuje funkcje systemu operacyjnego opisane w podrozdziałach 4.2. i 4.3.

4.5. SYSTEMY WIELOZADANIOWE I WIELODOSTĘPNE

Złożone systemy komputerowe (np. *main frame*) mogą wykonywać wiele programów równocześnie. Procesor lub częściej procesory wykonują jednocześnie kilka programów jednego użytkownika i wtedy mówimy o wielozadaniowości (ang. *multitasking*) lub programy kilku użytkowników, z których każdy ma własną końcówkę (*terminal*), zawierającą co najmniej klawiaturę i monitor, i wtedy mówimy o wielodostępnie (ang. *multiuser*

system). W obu przypadkach procesory wykonują na przemian fragmenty wielu programów, a dzięki szybkości pracy procesora użytkownicy mają wrażenie, że programy działają jednocześnie.

W przypadku wielozadaniowości system operacyjny wykonuje wiele skomplikowanych czynności. Przede wszystkim ładuje kolejne programy z pamięci dyskowej do pamięci operacyjnej, przydziela im odpowiednie jej bloki, a także urządzenia zewnętrzne. Ponadto, chroni programy i pliki przed niepowołanym dostępem, ustawia w kolejce programy oczekujące na wykonywanie, następnie przydziela im procesor, synchronizuje ich pracę itp. Takie systemy operacyjne wymagają dużej pamięci. W pamięci operacyjnej np. RAM umieszcza się więc tylko te fragmenty systemu i programów użytkowych, które są w danej chwili niezbędne lub wykonywane bardzo często, pozostałe części rezydują w pamięci dyskowej.

4.6. JĘZYKI PROGRAMOWANIA – KOMPILATORY I INTERPRETERY

W tym podrozdziale omówione zostaną języki związane z komputerami oraz ich hierarchia. Ponieważ procesor „rozumie” jedynie ciągi zer i jedynek, rozkazy można mu wydawać tylko w takim języku. Zbiór wszystkich kodów rozkazów zrozumiałych dla układu sterowania tworzy **język wewnętrzny** mikroprocesora zwany też **językiem maszynowym** lub językiem komputera. Rozkaz ma postać słowa maszynowego, składającego się z części operacyjnej (będącej zakodowanym działaniem) i części adresowej (informacja, na której to działanie ma być wykonane lub jej adres, tzn. numer komórki pamięci zawierającej potrzebną informację). W chwili gdy słowo znajduje się w rejestrze rozkazów, jest interpretowane jako rozkaz. Na wykonanie rozkazu składa się wiele operacji wykonywanych przez różne podzespoły komputera. Operacje te tworzą tzw. cykl rozkazu. Jedną z nich jest wskazanie słowa, które w następnym cyklu będzie wprowadzone do rejestru rozkazów jako następny wykonywany rozkaz. Przejście do następnego cyklu następuje zaraz po zakończeniu aktualnie realizowanego – w ten sposób przebiega samoczynne wykonanie sekwencji rozkazów, wymagające tylko zapoczątkowania (wprowadzenia do rejestru rozkazów pierwszego rozkazu). Zapisany w takim języku algorytm jest po prostu mało czytelnym ciągiem liczb – zer i jedynek. Jednak od czasu do czasu konieczne jest, zwłaszcza w zastosowaniach profesjonalnych, zapisanie algorytmu w języku wewnętrznym. W związku z tym tworzy się język wyższego poziomu, zastępując kody liczbowe języka komputera literowymi oznaczeniami oraz wprowadzając inne dogodne dla człowieka elementy. Otrzymany w ten sposób język nosi nazwę **języka symbolicznego**, zwanego też czasem językiem asemblera. Natomiast sam **assembler** jest programem, który tłumaczy program zapisany w języku symbolicznym na język wewnętrzny.

Najwyżej w hierarchii stoją **języki algorytmiczne**, nazywane też językami wysokiego poziomu. Do tej grupy należą takie znane języki jak ADA, ALGOL, COBOL, C++, FORTRAN, PASCAL, popularny kiedyś BASIC, a obecnie JAVA.

Program napisany w języku algorytmicznym, zwany kodem źródłowym, musi być przetłumaczony na język wewnętrzny, czyli na język komputera. Wykonuje to program kompilujący, czyli **kompilator**. Dopiero otrzymany po pracy kompilatora program w języku wewnętrznym, czyli kod wynikowy, może być wykonany przez komputer.

Kompilator analizuje program napisany w określonym języku programowania, czyli kod źródłowy i zamienia go na równoważny mu funkcjonalnie program w języku wewnętrznym procesora, czyli kod wynikowy. Jeśli proces ten zakończy się poprawnie, to w dalszej pracy używa się kodu wynikowego. Natomiast, gdy program (kod źródłowy) zawiera błędy, to cały proces tłumaczenia musi być powtórzony od początku po poprawieniu programu. Czas potrzebny na tłumaczenie zależy od języka, długości programu oraz jakości kompilatora i wynosi od kilku sekund do kilku godzin.

Kompilatory języków programowania wraz z programami je wspomagającymi grupowane są w biblioteki. Wyróżnione zostały dwie grupy bibliotek języków – maszynowe i problemowe. W pierwszej grupie zazwyczaj znajduje się kompilator języka wewnętrznego komputera oraz języków symbolicznych. W drugiej natomiast znajdują się kompilatory języków wysokiego poziomu lub języków specjalizowanych. Wyróżnia się tu ponadto kompilatory języków algorytmicznych, dialogowych oraz problemowych.

Drugą możliwość wykonania programu zapisanego w języku algorytmicznym daje program interpretujący, czyli **interpreter**. Interpreter tłumaczy kolejne instrukcje programu na sekwencję rozkazów komputera i natychmiast je wykonuje. Program nie jest tłumaczony i może być uruchomiony natychmiast. Niestety, ponieważ nie generuje się kodu wynikowego, to przy powtórnym wykonywaniu cały program (kod źródłowy) jest interpretowany na nowo. Programy interpretowane działają znacznie wolniej w porównaniu z programami skompilowanymi.

Kiedyś było tak, że interpretery używane w komputerach osobistych współpracowały z edytorami, które służyły do redagowania i poprawiania programów. Wówczas był to pierwszy program, z którym zapoznawał się użytkownik komputera.

4.7. POROZUMIEWANIE SIĘ Z KOMPUTEREM

Użytkownikami obecnie stosowanych powszechnie komputerów są zazwyczaj ludzie bez doświadczenia informatycznego. Wieloletnie wysiłki mające na celu opracowanie intuicyjnych metod komunikowania się zaowocowały standaryzacją wyglądu ekranu i podstawowych metod obsługi. Dzięki temu użytkownicy po krótkim treningu mogą posługiwać się oprogramowaniem w stopniu podstawowym.

W literaturze przedmiotu [1987] definiuje się programy przyjazne (ang. *user friendly*) jako programy, które powinny:

- komunikować się z użytkownikiem na zasadzie dialogu, na przykład, jeśli użytkownik ma podać dane, program powinien wyświetlić na ekranie odpowiedni komunikat,
- sprawdzać poprawność wprowadzanych danych i rozkazów, program akceptuje tylko poprawne dane i rozkazy, błędne nie powodują niepoprawnego działania programu,
- wyświetlać na ekranie wszystkie informacje potrzebne użytkownikowi do pracy; zwykle wyświetla się resume tych informacji w wydzielonej części ekranu; użytkownik po naciśnięciu odpowiedniego klawisza uzyskuje na ekranie dalsze informacje (ang. *help*), np. instrukcje dotyczące sposobu pracy, dostępnych opcji (możliwości do wyboru), znaczenia klawiszy itd.,

- na życzenie podpowiadać, jakie czynności należy wykonać, by zrealizować określone zadanie,
- wyświetlać na ekranie spis zadań, które mogą być wykonane, w postaci ich listy, tzw. menu.

Obecnie programy użytkowe na ogół realizują wszystkie powyższe funkcje. Współczesne komputery wyposażone są w system operacyjny o graficznym sposobie komunikowania się z użytkownikiem.

Dodatkowym elementem porozumiewania się użytkownika z komputerem są klawisze funkcyjne, np. duże/małe litery, F1 – pomoc, klawisze nawigacji (np. ze strzałkami) służące przesuwaniu kursora itp.

Być może wkrótce będziemy mogli porozumiewać się z komputerem za pomocą głosu i języka naturalnego.

4.7.1. POLECENIA

W latach osiemdziesiątych XX wieku użytkownik komputera osobistego porozumiewał się z programami poprzez polecenia wprowadzane za pomocą klawiatury, nazywane także dyrektywami lub komendami. Programy takie będziemy nazywać sterowanymi poleceniami (ang. *command driven*). Ta metoda komunikowania wymaga od użytkownika znajomości poleceń oraz ich parametrów, np. w komendzie drukowania należało podać parametr określający urządzenie, na którym realizowany był wydruk – drukarka lub monitor.

4.7.2. MENU

Dużo łatwiej jest korzystać z programów sterowanych za pomocą menu. Na ekranie program wyświetla listę pozycji do wyboru. Użytkownik wybiera element menu i albo jest on wykonywany, albo na ekranie pojawia się lista dla menu niższego poziomu. Pozycje z menu wybiera się, wskazując je kursorem, wciskając jeden z klawiszy wskazanych w menu lub korzystając z manipulatora – myszki.

W obecnych programach na ogół stosuje się menu rozwijane w dół (ang. *pull down menu*). Po raz pierwszy zastosowano je w komputerze Macintosh. Po wybraniu elementu z wyświetlanej poziomo w górnej części ekranu listy rozwija się lista z umieszczonymi pionowo elementami do wyboru.

Ważny jest właściwy dobór terminologii i struktury. Współczesne oprogramowanie ma hierarchiczną, wielopoziomą strukturę menu. W systemie operacyjnym Windows 98 menu „Start” rozwija się do góry w postaci listy pionowej, a podmenu rozwija się pionowo z prawej.

Menu może być zastąpione przez małe rysunki (ikony, ang. *icons*), które symbolizują dostępne polecenia lub opcje. Przesunięcie kursora na ikonę i naciśnięcie odpowiedniego klawisza myszki powoduje wykonanie symbolizowanej przez rysunek akcji.

4.7.3. ZAPYTANIA

Innym sposobem porozumiewania się oprogramowania z użytkownikiem są zapytania. Stosuje się je zazwyczaj wtedy, gdy obsługa programu wymaga podawania wielu danych. Komputer wyświetla są na ekranie pytania, użytkownik wprowadza odpowiedzi za po-

mocą klawiatury. Czasami wyświetlana jest podpowiedź, która użytkownik może potwierdzić, wciskając klawisz (Enter). W systemach do przetwarzania danych zapytania grupuje się w formularze zawierające wiele okienek (pól) do wypełnienia.

4.7.4. OKNA

Współczesne komputery porozumiewają się z użytkownikiem prawie wyłącznie za pomocą tzw. okien (ang. *windows*). Okno jest to wyróżniona prostokątna część ekranu, na ogół obwiedziona ramką, w której można przeprowadzać pewne operacje bez naruszania pozostałej części ekranu. W wielu programach wielkość okna i jego położenie na ekranie mogą być ustalane przez użytkownika. Okna wywoływane chwilowo na ekran zasłaniają częściowo poprzednią zawartość ekranu, ale jej nie zmieniają. Po zamknięciu okna ekran powraca do poprzedniego stanu lub zmieniany jest zgodnie z wyborem dokonany przez użytkownika w menu. W MS Windows istnieje możliwość kolejnego wywoływania wielu nakładających się na siebie okien. Okna można przesuwac po ekranie, powiększac, zmniejszac i zamykac, na ogół takze zasłaniają się częściowo.

Systemy wielozadaniowe zawsze używają okien. Każdy z wykonywanych programów używa własnego okna.

Przy współczesnych, graficznych systemach wizualizacji okna mogą mieć bardzo różne, czasem wymyślne kształty, tak jest np. w programach do odtwarzania plików muzycznych.

4.8. PRZEMYSŁ OPROGRAMOWANIA

Prawdopodobnie za pierwszy produkt przemysłu oprogramowania komputerów osobistych należy uznać opracowany w 1978 r. przez D. Bricklina, a wprowadzony na rynek w 1979 r. program VisiCalc na komputer Apple II. Ten pierwszy arkusz kalkulacyjny umożliwił użytkownikom bez wiedzy informatycznej planowanie, zapisywanie i wykonywanie złożonych obliczeń. W tym samym czasie (1978) powstał jeden z pierwszych programów redagowania tekstów: Wordstar firmy MicroPro. Jego autorem jest Robert Barnaby, a przeznaczony był on na komputery pracujące w systemie operacyjnym CP/M.

W USA najpierw głównymi odbiorcami programów dla komputerów klasy PC były osoby prywatne lub właściciele małych firm. Osoby te w wyborze kierowały się własnymi potrzebami zawodowymi oraz możliwością modyfikacji. Obecnie największe znaczenia ma rynek wielkich przedsiębiorstw, w których komputery osobiste włączane są do sieci informatycznej. Ze względu na dużą liczbę użytkowników i wysoki koszt zakupu wymaga to standaryzacji oprogramowania.

Przy wprowadzaniu komputerów osobistych do zakładów pracy bardzo ważnym elementem stała się dokumentacja i łatwość obsługi. Zmniejsza to koszty szkolenia pracowników. Popularność programu nie zawsze zależy od jego jakości lub ceny. Często decyduje termin jego wprowadzenia. Dlatego spotykamy się z reklamą programów, których nie ma jeszcze w sprzedaży. Taka reklama służy do powstrzymania ewentualnych nabywców przed zakupem już dostępnego programu innego producenta. W pogoni za konkurencją firmy wprowadzają nowe wersje bez ich dokładnego przetestowania i użytkownicy napotykać wiele usterek i błędów.

Większość znanych dziś na rynku przedsiębiorstw produkujących oprogramowanie powstała jako drobne firmy utworzone przez jednego lub kilku założycieli. Często byli to autorzy pierwszych wersji programów sprzedawanych przez te firmy. Niektóre z nich to obecnie wielkie przedsiębiorstwa zatrudniające setki programistów i oferujące szeroką gamę oprogramowania.

Dla powodzenia programu ważny jest nie tylko dobry, oryginalny pomysł, ale też oprawa programu, czyli wygląd ekranu i sposób porozumiewania się z użytkownikiem, współpraca z różnymi zestawami sprzętowymi, liczba realizowanych funkcji. Obecnie duże znaczenie ma reklama o szerokim zasięgu, umiejętnie ustalenie ceny, umieszczenie programu w sieci sprzedaży.

Wysokie ceny są nadal przyczyną dość powszechnego zjawiska nielegalnego kopiowania programów. Kiedyś producenci oprogramowania stosowali, z miernym skutkiem, wyrafinowane techniki zabezpieczenia przed nielegalnym kopiowaniem, ponosząc wysokie koszty, a to nie sprzyjało obniżaniu ceny. Obecnie stosuje się przede wszystkim środki prawne.

Niezależnie od rynku komercyjnego istnieje szeroki niekomercyjny przepływ programów organizowany przez użytkowników, w szczególności autorów programów określany jako *public domain*. Ogólnie *public domain*, czyli dobro publiczne, to ogół tekstów, zdjęć, muzyki, dzieł sztuki, oprogramowania itp., które na skutek decyzji twórcy, braku spadkobierców lub upływu odpowiedniego czasu stały się dostępne do dowolnych zastosowań bez ograniczeń wynikających z przepisów prawa autorskiego.

Jest wiele dobrych programów opracowanych przez niezawodowych autorów, którzy zawarli w nich swoje doświadczenie z danej dziedziny. Występują tu dwie formy: *freeware* – program darmowy i można go kopiować bez wprowadzania zmian, lub *shareware* – po wniesieniu niewielkiej opłaty użytkownik otrzymuje pełną wersję programu albo kod aktywujący posiadany już program.

Freeware to przede wszystkim licencja oprogramowania umożliwiająca darmowe rozprowadzanie aplikacji bez ujawnienia kodu źródłowego, choć bywają wyjątki. Termin *freeware* bywa również używany jako synonim oprogramowania objętego tą licencją. Czasami licencja *freeware* zawiera dodatkowe ograniczenia, np. część *freeware* jest całkowicie darmowa, a część darmowa jedynie do użytku nie związanego z działalnością gospodarczą.

Shareware jest płatnym programem komputerowym, który jest rozpowszechniany bez opłat do sprawdzenia przez użytkowników. Po okresie próbnym za taki program należy zapłacić lub skasować go. Używanie programu *shareware* po okresie próbnym bez opłaty jest nielegalne.

W ostatnich latach pojawiła się jeszcze jedna kategoria oprogramowania. *Adware* to typ licencji zamkniętego oprogramowania. Oprogramowanie *adware* jest rozpowszechniane za darmo i podobnie jak *freeware* bez kodu źródłowego, lecz zawiera funkcję wyświetlającą reklamy, na których zarabia wydawca oprogramowania. Zwykle jest też możliwość nabycia za opłatą wersji programu bez reklam.

Obecnie głównym środkiem rozpowszechniania tego typu oprogramowania jest Internet. Mimo upływu lat trudno się dziś nie zgodzić z wyrażonym w literaturze [1988] poglądem, że:

Przemysł komputerowy, a zwłaszcza przemysł oprogramowania, jak żadne inne, oparte są na wiedzy i – co warto podkreślić – zdolnościach ludzi młodych. Steve Jobs i Steve Wozniak – twórcy komputera Apple, David Bricklin – autor programu VisiCalc, Phillip Kahn – założyciel firmy Borland i dziesiątki innych osiągnęli swe sukcesy w wieku poniżej 30 lat. Do tego należy dodać, że oprogramowanie komputerów osobistych nie jest wyłączną domeną zawodowych informatyków. Każda dziedzina zastosowań wymaga przede wszystkim wiedzy z tej dziedziny: ekonomicznej, medycznej czy inżynierskiej. Pomimo wielkiego postępu daleko jest jeszcze do wyczerpania możliwości doskonalenia programów. Nawet najlepsze istniejące dziś programy mają wiele ograniczeń.

4.9. APLIKACJE

Aplikacja według internetowego *Słownika języka polskiego* to *3. ozdobny wzór wycięty z tkaniny lub skóry naszyty na tło z innej tkaniny; też technika haftu polegająca na naszywaniu takich wzorów*. Jest to dobre wprowadzenie do rozmytej definicji programu użytkowego, w której aplikacja to rodzaj programu komputerowego niezaliczanego do systemu operacyjnego; program ten ma bezpośredni kontakt z użytkownikiem i nie jest częścią większego programu. Z technicznego punktu widzenia jest to oprogramowanie korzystające z usług systemu operacyjnego.

Przykłady aplikacji to menedżery plików, oprogramowanie biurowe (redaktory tekstu, arkusze kalkulacyjne, programy finansowe) oraz gry, ale także strony WWW napisane z użyciem języków PHP, ASP, JSP i innych.

Bazą dla rozwoju oprogramowania komputerów osobistych była teoria i praktyka programowania dla dużych komputerów popularnych w latach sześćdziesiątych i siedemdziesiątych XX wieku. Przyspieszenie rozwoju nastąpiło, gdy zdano sobie sprawę, że komputer osobisty jest nowym narzędziem. Obecnie programy użytkowe przygotowane dla komputerów osobistych nazywamy aplikacjami.

Współcześni użytkownicy komputerów osobistych nie muszą znać języków programowania. Dzieci rozpoczynają zwykle kontakt z komputerem od gier, nabierają wprawy w operowaniu klawiaturą i manipulatorami. Poznają również gry strategiczne lub symulacyjne, które uczą podejmowania decyzji i optymalizacji działań.

Nie należy się temu dziwić, gdyż jeżeli program ma być „przyjazny” dla użytkownika, to musi spełniać wiele czasem trudnych do pogodzenia wymagań. Nie usprawiedliwiam w ten sposób producentów oprogramowania, gdyż w pogoni za nowymi wersjami – czytając zyskami – skracają czas testowania, przerzucając ten trud na tych, którzy pierwsi kupią ich nowe produkty.

Współczesna klawiatura komputera klasy IBM PC ma 12 klawiszy funkcyjnych, które mogą być wykorzystane w aplikacjach do wyboru różnych specyficznych operacji. Korzystając z możliwości równoczesnego użyciu klawiszy sterujących „Ctrl”, „Shift” lub „Alt”, można zdefiniować prawie 50 różnych funkcji. Przyjęto, że klawisz funkcyjny „F1” służy do wyświetlenia pomocy lub podpowiedzi. Standaryzacja znaczenia klawiszy korzystnie wpływa na sprawność posługiwania się oprogramowaniem użytkowym. Na przykład w aplikacjach pracujących w systemie operacyjnym MS Windows, niezależnie od wersji, zestandaryzowano znaczenie kombinacji klawiszy „Ctrl+C”, „Ctrl+V”, „Ctrl+X”.

Współczesne aplikacje umożliwiają użytkownikowi zapamiętanie sekwencji klawiszy i związanych z nimi operacji w postaci makroinstrukcji (makrorozkazu). Tworzenie makroinstrukcji jest równoznaczne z programowaniem, a symbole rozkazów i struktura listy rozkazów tworzą składnię specyficznego języka programowania typu interpretacyjnego. W niektórych aplikacjach, na przykład MS Word, MS Excel, użytkownik może zaprogramować złożone operacje w tzw. trybie rejestracji (uczenia). Wówczas program śledzi kolejne wydawane przez użytkownika rozkazy i zapamiętuje je jako makroinstrukcję. Im większa jest liczba podstawowych operacji w programie, tym większa jest jego elastyczność, ale oczywiście specjalizowane programy użytkowe nie dają takich możliwości programowania, jak uniwersalne języki programowania.

Ze względu na wysoki koszt druku obszernych podręczników oraz szybką zmianę wersji programy wyposażane są w rozbudowany system pomocy. Aplikacja ma zatem wbudowany skrócony podręcznik użytkownika. Na ogół informacja wyświetlana na ekranie odpowiada potrzebie chwili, to znaczy wyjaśnia działania, które użytkownik może podjąć w określonej sytuacji. Ponadto istnieje możliwość sięgnięcia przez użytkownika do dowolnych innych informacji o programie. Informacja wywoływana z programu nosi nazwę „pomocy bieżącej” (ang. *context sensitive help*). Często aplikacje uzupełniane są o specjalne programy instruktażowe (ang. *tutorials*), które krok po kroku uczą podstawowego stosowania programu.

4.10. POWSTAWANIE SYSTEMÓW KOMPUTEROWYCH I OPROGRAMOWANIA*

W tym miejscu zasygnalizujemy tylko problem tworzenia złożonych systemów i programów komputerowych. Firmy produkujące oprogramowanie opracowały własne systemy pracy i kontroli wytwarzania tych produktów zwane metodykami. Istnieją w tym zakresie zarówno ogólne standardy, jak i specyficzne dla danego producenta rozwiązania. Ogólnie możemy powiedzieć, że tworzenie zarówno systemów, jak i oprogramowania składa się z następujących etapów:

1. Analiza potrzeb,
2. Projektowanie,
3. Programowanie,
4. Testowanie.

Oczywiste jest, że te etapy służą nie tylko budowaniu, ale są także stosowane podczas rozwijania oprogramowania. Etapy pozwalają w sposób systematyczny kontrolować tworzenie i rozwój oprogramowania. Firmy stosują własne metodyki, jednak wszystkie zawierają powyższe elementy.

* Zainteresowanych metodyką realizacji projektów odsyłam do książki *Kompendium wiedzy o zarządzaniu projektami. PM BOK Guide 2000 Edition*, Warszawa 2003, s. 20–22, *Tworzenie oprogramowania*.

4.1 1. ETAPY ROZWOJU UMIEJĘTNOŚCI STOSOWANIA KOMPUTERA

W literaturze [1988] znajdujemy klasyfikację umiejętności posługiwania się komputerem i językami programowania (autorzy cytują ją za P. Jarvinen, *Convention Informatique* 1985, 16–20 września 1985 r., Paris, tom A, s. 455).

1. *Użytkownik zna ogólnie możliwości i ograniczenia komputera. Może brać udział w formułowaniu zadań dla systemu.*
2. *Użytkownik zna dostępne dane i programy oraz wie, czego konkretnie może zażądać od eksperta obsługującego komputer.*
3. *Użytkownik umie uruchomić program.*
4. *Użytkownik może sterować wykonywaniem programu.*
5. *Użytkownik umie programować.*
6. *Użytkownik umie tworzyć narzędzia programowania.*

Zauważmy, że klasyfikacja powstała w okresie dominacji komputerów dużych i mimo upływu lat nie straciła wiele na aktualności. W punkcie 2 użytkownik sam jest ekspertem obsługującym komputer. W 1985 r. obliczenia wykonywane były przede wszystkim na dużych komputerach. Rozwiązywanie problemów wymagało opisanie ich w jakimś języku programowania oraz przygotowania ciągu poleceń dla systemu operacyjnego. Użytkownik musiał albo sam umieć programować albo korzystać z pomocy programisty – eksperta komputerowego. W tamtych latach komputery osobiste były drogie i rzadkie. Była to wówczas nowość. Punkt 3. stał się trywialny, gdyż obecnie zazwyczaj wystarczy kliknięcie na ikonie. Niestety, nie zawsze, a zatem punkt 4. Pozostałe dwa punkty pozostawiamy wykształconym informatykom, a punkt 5 ewentualnie jeszcze hobbystom.

4.1 2. PRAWO AUTORSKIE**

W Polsce terminem **prawo autorskie** określa się zarówno ogół praw przysługujących autorowi dzieła, jak i przepisy upoważniające autora do decydowania o użytkowaniu dzieła i czerpaniu z niego korzyści. W ustawie „O prawie autorskim i prawach pokrewnych” z dnia 4 lutego 1994 r. (tekst jednolity: Dz. U. nr 80 z 2000 r., poz. 904) z późniejszymi zmianami wyróżnia się autorskie prawa osobiste oraz autorskie prawa majątkowe.

Zgodnie z Art. 1 Ustawy **przedmiotem prawa autorskiego** jest każdy przejaw działalności twórczej o indywidualnym charakterze, ustalony w jakiejkolwiek postaci, niezależnie od wartości, przeznaczenia i sposobu wyrażenia (utwór). W szczególności przedmiotem prawa autorskiego są utwory:

- wyrażone słowem, symbolami matematycznymi, znakami graficznymi (literackie, publicystyczne, naukowe, kartograficzne oraz **programy komputerowe**),
- plastyczne,
- fotograficzne,

**Więcej informacji czytelnik znajdzie w specjalistycznych publikacjach z zakresu prawa autorskiego, np. J. Barta, R. Marwickiewicz, *Prawo autorskie. Przepisy, orzecznictwo, umowy międzynarodowe*, Warszawa 2002.

- lutnicze,
- wzornictwa przemysłowego,
- architektoniczne, architektoniczno-urbanistyczne i urbanistyczne,
- muzyczne i słowno-muzyczne,
- sceniczne, sceniczno-muzyczne, choreograficzne i pantomimiczne,
- audiowizualne (w tym wizualne i audialne).

Utwór jest przedmiotem prawa autorskiego od chwili ustalenia, chociażby miał postać nieukończoną. Ochrona przysługuje twórcy niezależnie od spełnienia jakichkolwiek formalności.

Autorskie prawa majątkowe mogą przejść na inne w drodze dziedziczenia lub na podstawie umowy. Umowa o przeniesienie autorskich praw majątkowych lub umowa o korzystanie nazywana jest licencją.

Licencja – dokument prawny lub umowa, określająca warunki korzystania z utworu jako dobra, którego dana licencja dotyczy. Właściciel praw autorskich, znaku handlowego lub patentu może (i często to robi) wymagać od innych posiadania licencji jako warunku użytkowania lub reprodukcji licencjonowanego utworu. Wyróżnia się licencje: pełną, wyłączną, niewyłączną, otwartą, przymusową oraz sublicencję.

Nie rozważając w sposób ogólny rodzajów licencji, skupimy się na rodzajach licencji na oprogramowanie. Rozdział 7. Ustawy zawiera przepisy szczegółowe dotyczące programów komputerowych.

Licencja na oprogramowanie to umowa na korzystanie z utworu, jakim jest program/aplikacja komputerowa, zawierana pomiędzy podmiotem, któremu przysługują majątkowe prawa autorskie do utworu, a osobą, która zamierza z danej aplikacji korzystać.

Umowa taka powinna określać przede wszystkim pola eksploatacji (formy wykorzystania utworu), czyli warunki, na jakich licencjobiorca jest uprawniony do korzystania z programu.

Licencje niewyłączne – udzielający licencji może zezwolić na korzystanie z utworu wielu osobom równocześnie; licencja ta nie musi mieć formy pisemnej.

Licencje wyłączne – (rzadkość w przypadku aplikacji komputerowych), spotykane głównie w przypadku znacznych kontraktów, na przykład na wykonanie systemów bankowych itp., w tym przypadku zwykle umowa licencyjna wynika z umowy o dzieło, na której podstawie firma wykonująca oprogramowanie wykonuje zamówioną aplikację, umowa taka wymaga formy pisemnej pod rygorem nieważności,

Sublicencja – licencjobiorca może udzielić dalszej licencji, pod warunkiem wszakże takiego upoważnienia w jego umowie licencyjnej.

Dla zrównoważenia ochrony praw majątkowych wprowadzonych przede wszystkim przez przemysł oprogramowania zaproponowano licencję chroniącą autorskie prawa osobiste, lecz umożliwiającą nieodpłatne i legalne użytkowanie programów.

Powszechna Licencja Publiczna GNU (General Public License) jest jedną z licencji wolnego oprogramowania, która została sformułowana w 1988 przez Richarda Stallmana i Ebena Moglena na potrzeby Projektu GNU, na podstawie wcześniejszej Emacs General Public License. Wersja 2 licencji GNU GPL została wydana w roku 1991.

Celem tej licencji jest przekazanie użytkownikom praw do uruchamiania programu w dowolnym celu (wolność 0), analizowania działania programu i dostosowywania go do swoich potrzeb (wolność 1), kopiowania (wolność 2) oraz udoskonalania i publikowania

własnych poprawek programów i kodu źródłowego tych programów, których programiści postanowili wydać je na podstawie tej licencji (wolność 3).

W zakresie oprogramowania najczęstsze ograniczenia licencyjne, to:

- ograniczenie liczby kopii użytkowanego oprogramowania,
- zakaz udostępniania programu innym użytkownikom.

Podstawowe zasady związane z prawem autorskim, a dotyczące oprogramowania:

- programy komputerowe są chronione prawem autorskim,
- nielegalne kopiowanie programów jest kradzieżą,
- prawo do korzystania z programu komputerowego nabywa się od autora, producenta lub upoważnionego ich przedstawiciela, tzn. właściciela praw majątkowych,
- oprócz praw osobistych autor posiada także prawa majątkowe, które umożliwiają mu czerpanie zysku materialnego ze swej pracy twórczej.

4.1 3. BEZPIECZEŃSTWO I OCHRONA DANYCH

Mówiąc o bezpieczeństwie i ochronie danych, musimy zwrócić uwagę na dwa aspekty: fizyczny i prawny.

Źródłem zagrożeń jest w szczególności pobieranie danych z Internetu, korzystanie z poczty elektronicznej oraz używanie i rozpowszechnianie informacji zawartych na nośnikach, takich jak CD-ROM, dyski Zip, dyskietki. Inne zagrożenie to awaria techniczna, na przykład uszkodzenie dysku twardego bądź nośnika, na którym zapisaliśmy dane. W każdej sytuacji ważne jest tworzenie **kopii bezpieczeństwa**, czyli zapisywanie danych na innych nośnikach, na przykład CD-ROM-ach lub dodatkowych dyskach.

Ochroną profilaktyczną przed zagrożeniami z Internetu i poczty elektronicznej, a ogólnie mówiąc, z sieci komputerowej jest posiadanie aktualizowanej ochrony antywirusowej i zabezpieczenia przed niepożądanym dostępem, na przykład w postaci ściany ogniowej (ang. *firewall*) prywatnej bądź firmowej.

W zakresie ochrony prawnej danych mamy do dyspozycji pewne fragmenty prawa autorskiego, chciałbym jednak zwrócić uwagę na Ustawę o ochronie danych osobowych. Jej znaczenie trudno przecenić ze względu na szeroki zakres stosowania komputerowych baz danych i związaną z tym łatwością ich przenoszenia bądź rozpowszechniania. Dane osobowe wykorzystuje się m.in. w przedsiębiorstwach w celu ewidencji pracowników, ale także w celach marketingowych.

Ochrona danych osobowych^{***} to regulacje prawne mające na celu ochronę prawa każdego do prywatności, a w szczególności jego danych osobowych przetwarzanych przez różne instytucje i przechowywane w ich bankach danych. Ujmują to trzy podstawowe akty prawne:

- Ustawa z 29 sierpnia 1997 o ochronie danych osobowych.
- Rozporządzenie Ministra Spraw Wewnętrznych i Administracji z 29 kwietnia 2004 w sprawie dokumentacji przetwarzania danych osobo-

^{***} Czytelnik może się zapoznać z komentarzem np. w książce: J. Barta, P. Fajgielski, R. Markiewicz, *Ustawa o ochronie danych osobowych - komentarz*, wyd. III, 2004.

wych oraz warunków technicznych i organizacyjnych, jakim powinny odpowiadać urządzenia i systemy informatyczne służące do przetwarzania danych osobowych.

- Rozporządzenie ministra spraw wewnętrznych i administracji z 29 kwietnia 2004 w sprawie wzoru zgłoszenia zbioru danych do rejestracji Generalnemu Inspektorowi Ochrony Danych Osobowych.

W rozumieniu Ustawy, zgodnie z art. 6., za dane osobowe uważa się wszelkie informacje dotyczące zidentyfikowanej lub możliwej do zidentyfikowania osoby fizycznej. W szczególności może to być numer identyfikacyjny albo jeden lub kilka specyficznych czynników określających cechy fizyczne, fizjologiczne, umysłowe, ekonomiczne, kulturowe lub społeczne.

W rozdziale 3 Ustawy określono zasady przetwarzania danych osobowych. Przetwarzanie jest dopuszczalne, gdy:

- osoba, której dane dotyczą, wyraża na to zgodę,
- jest to niezbędne do zrealizowania uprawnienia lub spełnienia obowiązku wynikającego z przepisu prawa,
- jest to konieczne do realizacji umowy, gdy osoba jest jej stroną,
- jest niezbędne do wykonania zadań realizowanych dla dobra publicznego,
- jest niezbędne do wypełniania prawnie usprawiedliwionych celów administratorów danych, nie naruszając praw i wolności osoby, której dane dotyczą.

Ustawodawca określił, że za prawnie usprawiedliwiony cel uważa się w szczególności marketing bezpośredni własnych produktów i usług administratora danych oraz dochodzenie roszczeń z tytułu prowadzonej działalności gospodarczej.

Administrator danych osobowych ma obowiązek poinformować osobę, której dane dotyczą o:

- adresie swojej siedziby i pełnej nazwie,
- informacji o celu zbierania danych i ewentualnie potencjalnych odbiorcach danych,
- prawie dostępu do danych i ich poprawiania.
- W rozdziale 4 Ustawy określono prawa osoby, której dane dotyczą, a w szczególności prawo do:
 - uzyskania informacji o istnieniu zbioru danych i jego administratorze wraz z siedzibą i pełną nazwą,
 - informacji o celu zbierania danych, zakresie i sposobie ich przetwarzania,
 - informacji, od kiedy dane są przetwarzane i przekazanie w powszechnie zrozumiałej formie tych danych,
 - uzyskania informacji o źródle, z którego pochodzą jego dane,
 - informacji o sposobie udostępniania danych, w szczególności o ich odbiorcach lub kategoriach odbiorców,

- żądania uzupełnienia, uaktualnienia i sprostowania danych osobowych oraz czasowego zaprzestania ich przetwarzania,
- wniesienia żądania zaprzestania przetwarzania danych zainteresowanej osoby,
- wniesienia sprzeciwu wobec przetwarzania danych zbieranych w innych celach, do celów marketingowych oraz wobec udostępnienia zgromadzonych danych innym administratorom.

Oprócz przestrzegania zasad przetwarzania danych osobowych i praw osób, których dane znajdują się w bazie, administrator danych osobowych (ADO) ma obowiązek ich zabezpieczania zgodnie z rozdziałem 5 Ustawy i przytoczonym wyżej rozporządzeniem. Ponadto, jeżeli Ustawa nie stanowi inaczej, ma obowiązek zarejestrować zbiór danych osobowych. Proszę zwrócić uwagę, że obowiązek rejestracji nie zależy od sposobu przetwarzania. Dotyczy zarówno zbiorów kartotekowych (papierowych), jak i komputerowych.

Rozdział 8 Ustawy określa konsekwencje karne, jakie poniesie ADO, jeżeli będzie przetwarzał dane niezgodnie z przepisami Ustawy, a w szczególności dopuści do ich ujawnienia osobom lub instytucjom nieuprawnionym lub nie poinformuje osoby, której dane dotyczą o ich przetwarzaniu.

5. MULTIMEDIA

Multimedia to ogólne określenie środków komunikacji wykorzystujących wiele form przekazu. Jeśli ktoś mówi o multimediami, to zazwyczaj ma na myśli połączenie kilku z następujących elementów:

- tekst,
- obraz (statyczny – grafika/zdjęcie; ruchomy – film/animacja),
- dźwięk (muzyka, mowa).

Dodatkowo, jeśli multimedia pojawiają się w kontekście komputerów, do grupy wymienionych elementów dochodzi wykorzystanie hiperłączy.

Hiperłącze, inaczej **łącznik hipertekstowy** (odsylacz, odnośnik, ang. *hyperlink*), to element dokumentu elektronicznego (tekstowego, graficznego, wideo, animacji, PDF, HTML) umożliwiający przejście do innego miejsca w danym dokumencie lub do innego dokumentu.

Sama koncepcja hiperłącza związana z pojęciem hipertekstu i jako idea pojawiła się w pracach Paula Otleta przed II wojną światową. Rozwijana sukcesywnie stała się podstawą stron internetowych. Hiperłącza umożliwiają budowę złożonych dokumentów odwołujących się do różnych źródeł. W edycji tekstu pojawiło się po raz pierwszy w redaktorze tekstu WordPerfect w wersji 6 dla DOS na początku lat 90. XX w. Obecne redaktory tekstu w pakietach biurowych poprzez hiperłącza umożliwiają tworzenie odwołań, na przykład od pozycji spisu treści, skorowidza lub wykazu ilustracji do odpowiednich miejsc w treści dokumentu. Dotyczy to również innych elementów pakietów biurowych lub aplikacji, a w szczególności dokumentów zawartych na stronach internetowych.

5.1. URZĄDZENIA TECHNICZNE

Rozważając zastosowania komputerów w przetwarzaniu obrazów, zaczniemy od urządzeń technicznych niezbędnych do pokazania obrazu i wprowadzenia go do komputera. Obraz, zdjęcie, grafikę możemy wprowadzić do komputera na cztery sposoby:

- za pomocą skanera,
- za pomocą aparatu cyfrowego,
- za pomocą kamery cyfrowej,
- za pomocą programu do rysowania ilustracji.

Skanery występują w dwóch rozwiązaniach – płaskie i bębnowe. Pierwsze są na ogół formatu A4 i mogą być podłączone do komputera na trzy sposoby – tańsze przez port drukarki (LPT) lub złącze USB, droższe przez kartę SCSI. Prędkości skanowania przy podłączeniu przez złącze USB i port SCSI są porównywalne i większe na ogół niż przez port LPT. Skanowanie odbywa się w kolorze, a sprzętowa rozdzielczość skanowania wynosi obecnie na ogół 600–4800 dpi. Skanery bębnowe zaliczane są na ogół do rozwiązań profesjonalnych i mogą skanować duże formaty.

Aparaty i kamery cyfrowe umożliwiają wprowadzenie obrazu cyfrowego do komputera bez konieczności zamiany na postać cyfrową. Jest tu cała gama rozwiązań wielu czołowych producentów sprzętu fotograficznego i elektronicznego, a także komputerowego (Olympus, Sanyo, Agfa, Minolta, Hewlett-Packard, Canon).

Przedstawmy trochę danych o monitorach i sterownikach graficznych. Standardowy monitor komputerowy, typu CRT (ang. Cathode Ray Tube), działa na podobnej zasadzie jak telewizor. Podstawowymi parametrami monitora są rozmiar, czyli przekątna ekranu i rozdzielczość. Obecnie spotyka się monitory kolorowe o przekątnej 15", 17", 19" i 21". Wcześniej były monitory 12" i 14", zarówno kolorowe, jak monochromatyczne. Konkurencją dla monitorów CRT są monitory płaskie o ekranach zrobionych z ciekłych kryształów, oznaczane jako LCD (Liquid Cristal Display), o przekątnej 15", 17" i 19".

Teraz krótka definicja: PIKSEL – zbitka skróconych wyrazów *picture* i *element*, najmniejszy element obrazu, jaki urządzenie (monitor) może wyświetlić.

Wyświetlanie obrazu na monitorze odbywa się za pomocą sterownika umieszczonego w komputerze zwanego kartą grafiki. W ciągu ostatnich dwudziestu lat w komputerach stosowano następujące typy sterowników: CGA, HGC, MDA, VGA i super VGA.

Karta grafiki kolorowej (CGA) umożliwiała wyświetlanie grafiki w 4 kolorach przy rozdzielczości 320 pikseli na 200 linii lub w dwóch kolorach w rozdzielczości 640 pikseli na 200 linii.

Karta grafiki monochromatycznej (MDA) wyświetlała wyłącznie tekst w jednym kolorze w rozdzielczości 720 pikseli na 320 linii, a znak tekstu miał rozmiary 7x9 pikseli.

Karta graficzna Hercules (HGC) umożliwia wyświetlanie w trybie monochromatycznym tekstu i grafiki w rozdzielczości 720 pikseli na 348 linii.

Udoskonalona karta graficzna (EGA) umożliwia wyświetlanie 16 kolorów z rozdzielczością 640 pikseli na 350 linii. Te 16 kolorów wybiera się z palety 64 barw.

Graficzna macierz video (VGA) jest standardem grafiki rastrowej wprowadzonym przez firmę IBM w 1987 r. Umożliwia wyświetlanie 256 kolorów z rozdzielczością 640 pikseli na 480 linii.

Karta super VGA jest udoskonaloną macierzą graficzną. Karta ta umożliwia wyświetlanie co najmniej 800 pikseli na 600 linii, a ponadto 1024x768, 1280x1024 w 16, 256, 65 536 lub 16,7 miliona kolorów jednocześnie. Przy dużej liczbie kolorów trzeba prawie 4 MB pamięci karty graficznej, aby wyświetlić obraz.

Istotnym elementem związanym z obrazem jest sposób tworzenia kolorów.

CMY jest to system otrzymywania kolorów w druku z trzech podstawowych farb: turkusowej (odcień niebieskiego – oznaczany literą C od ang. *cyan*), karmazynowej (odcień

czerwonego – M od ang. *magenta*) i żółtej (Y od ang. *yellow*). Kolor farby związany jest z częścią widma padającego na kartkę papieru białego światła, która nie jest przez farbę pochłonięta. Kolor żółty wydaje się żółty, ponieważ farba pochłania z padającego na kartkę światła białego wszystkie długości fal z wyjątkiem żółtej części widma, która jest odbita jak od lustra. Podstawowe farby CMY połączone przy maksymalnym nasyceniu dadzą kolor czarny – cała energia światła białego zostanie pochłonięta (można to wypróbować, mieszając na papierze niebieską, czerwoną i żółtą farbę, a otrzymany kolor będzie brązowo-czarny). Model CMY nazywany jest substraktywnym sposobem tworzenia barw, ponieważ usuwa podstawowe barwy RGB ze światła białego.

CMYK (Cyan-Magenta-Yellow-black) – model barw oparty na czterech kolorach: niebiesko-zielonym, karmazynowym, żółtym i czarnym. Jest w znacznym stopniu niezależny od sprzętu i stosuje się go głównie w drukarniach. Choć teoretycznie za pomocą zmieszania barw CMY można uzyskać kolor czarny, w praktyce otrzymuje się kolor z odcieniem brązowym. Aby obejść ten problem (w druku kolor czarny jest najważniejszy), rozbudowano model CMY o czwartą składową – czystą czarną barwę K (od ang. *black*). W programach komputerowych barwy modelu CMYK opisywane są podobnie jak barwy RGB liczbami od 0 do 255 każda i oznaczają procentowe pokrycie piksela drukowanego na papierze. Programy graficzne obsługują nie tylko model RGB, ale także model CMYK.

RGB (Red-Green-Blue) jest modelem barw opartym na trzech podstawowych kolorach – czerwonym, zielonym i niebieskim, w którym określony kolor definiowany jest przez podanie względnej zawartości każdego z kolorów podstawowych. Czerwony obszar na ekranie monitora wygląda czerwono, ponieważ emituje dużą porcję energii świetlnej w czerwonym fragmencie spektrum widzialnego. Kolory RGB przy maksymalnym nasyceniu tworzą kolor biały (możemy zaobserwować to na przykład w teatrze czy na koncercie, gdzie światło zmieszane z reflektorów: czerwonego, zielonego i niebieskiego jest białe). Podczas wyświetlania na monitorze używa się jednego bajtu do reprezentowania nasycenia koloru, na przykład 0,0,0 to kolor czarny, a 255,255,255 biały. Możemy wygenerować zatem $256 \times 256 \times 256 = 16\,777\,216$ barw RGB. Ten 24-bitowy sposób kodowania barw nosi nazwę True Color. Możliwe są oszczędniejsze sposoby kodowania barw dla 256 różnych kolorów (8-bitowy kolor) i 65 536 (16-bitowy kolor, High Color) oraz bardziej zaawansowane dla przyspieszenia odświeżania ekranu (32-bitowy kolor).

Zauważmy, że model CMY jest odwróconą wersją modelu RGB. Podstawowe barwy farb i podstawowe kolory monitora są matematycznie powiązane. Każde dwa podstawowe kolory RGB połączone dają jedną z podstawowych barw CMY. Każde dwie farby podstawowe zmieszane dają barwę jednego z podstawowych kolorów RGB.

✓ **HSB** (Hue-Saturation-Brightness) lub **HLS** (Hue-Luminosity-Saturation) – model opisywania kolorów na ekranie komputera poprzez cień, nasycenie i jasność.

5.2. CYFROWY ZAPIS OBRAZU

W graficznych środowiskach programów komputerowych niezbędne jest przechowywanie dużej liczby obrazowych danych graficznych. Przede wszystkim są to obrazy zbudowane z danych rastrowych (pikseli). Dane te, w zależności od karty graficznej, zajmują stosunkowo duże objętościowo obszary pamięci RAM. Na przykład obraz graficzny na karcie VGA (800x600 16 kolorów) to 240 KB pamięci, ale SVGA (800x600 16,7 milionów kolorów) to prawie 4 MB. Są to poważne obciążenia ilościowe, istotne przy przechowywaniu tych danych na dyskach twardych. Stąd potrzeba odpowiedniego zmniejszania liczebności takich danych. Powstała cała rodzina metod odpowiednio kodujących (kompresujących) dane graficzne i zapisujących je w odpowiednich formatach na pliku dyskowym. Popularnymi do dziś formatami danych graficznych, na ogół związanych ze środowiskami programów, dla których zostały po raz pierwszy przygotowane, są:

- BMP Microsoft WINDOWS – wszystkie wersje,
- GEM/IMG Ventura, GEM Art. Line,
- GIF CompuServe,
- JPG, JPEG Joint Photographic Expert Group – kompresja obrazu,
- LBM/IFF DeLuxePaint,
- MAC MacPaint,
- MSP MSP Paint,
- PCX PaintBrush,
- PIC PC Paint, Pictor,
- TIFF Aldus Page Maker,
- CDR CorelDraw
- WPG Word Perfect,
- PostScript Adobe Systems.

Wśród formatów graficznych najbardziej zaawansowany jest format TIFF (Tag Image File Format) przeznaczony do zapisu danych rastrowych. Ze względu na postępującą integrację środowisk programów graficznych stworzono bardziej zaawansowane formaty umożliwiające zapis danych rastrowych, wektorowych i wielu innych typów w jednym formacie, na przykład CGM (Computer Graphics Metafile).

Najprostszy sposób kodowania (kompresowania) danych obrazowych w plikach graficznych polega na zastąpieniu odpowiednio długich sekwencji powtarzających się danych sekwencją złożoną z danej oznaczającej początek sekwencji, danej odpowiadającej liczbie powielonych bajtów oraz danej, którą się powiela. Oczywiście, kodowanie ma sens tylko w wypadku, gdy liczba danych powtarzających się jest większa od długości sekwencji danych po zakodowaniu, a w opisanej metodzie jest większa od 3. Przykładami kodowania są formaty PIC, PCX i najbardziej zaawansowany JPEG.

Wspomniany już format plików TIFF (Tagged Information File Format) promowany przez firmy Aldus i Microsoft stał się przy przechowywaniu graficznych plików obrazowych jednym ze standardowych formatów.

Format ten jest stosowany przede wszystkim w trzech dziedzinach: Desktop Publishing z aplikacjami takimi jak PageMaker, InDesign czy Ventura, technikach rozpoznawania pisma (Optical Character Recognition), w których skaner służy jako urządzenie do

wprowadzania obrazu, a odpowiednie programy (Recognita, FineReader) do identyfikacji liter w tych obrazach, oraz w przesyłaniu siecią dokumentów, na przykład faksem. Standard formatu TIFF został tak pomyślany, aby ułatwić wymianę cyfrowych danych zawierających obraz.

Współcześnie priorytetem jest taka konstrukcja strukturalności danych, która umożliwi łatwe rozszerzenia formatu i właśnie TIFF to zapewnia. Podstawowym celem w projektowaniu formatu było dostarczenie bogatego środowiska umożliwiającego dowolną wymianę danych graficznych pomiędzy różnymi aplikacjami.

Format TIFF wykorzystywany jest we wszystkich rodzajach skanerów, urządzeniach przekazujących dokumenty poprzez sieć i większości programów przeznaczonych do obróbki obrazu. TIFF ma być w zamierzeniach twórców niezależny od środowiska informatycznego (komputera, systemu operacyjnego, aplikacji). Jedynym znaczącym założeniem jest, by medium dla przechowywania danych było podobne do pliku, tzn. było określone jako sekwencja bajtów ponumerowanych od 0 do N. Największym możliwym rozmiarem pliku TIFF jest 2^{32} bajtów.

Standardowym rozszerzeniem nazwy pliku dla formatu TIFF w systemach MS DOS oraz Windows 95 i 98 jest TIF, w środowisku Macintosha rozszerzenie jest w postaci TIFF.

Wiele programów aplikacyjnych potrafi odczytać obraz zapisany w pliku TIFF, ale nie wykorzystuje wszystkich przedstawionych możliwości formatu.

5.3. GRAFIKA KOMPUTEROWA

Grafika komputerowa jest działem informatyki zajmującym się tworzeniem obrazów i wizualizacją danych z wykorzystaniem komputerów. Grafika komputerowa stosowana jest powszechnie w nauce, technice, kulturze oraz rozrywce. Oto przykładowe zastosowania:

- kartografia,
- wizualizacja danych pomiarowych (np. w formie wykresów dwu- i trójwymiarowych),
- wizualizacja symulacji komputerowych,
- diagnostyka medyczna,
- kreślenie i projektowanie wspomagane komputerowo,
- przygotowanie publikacji,
- efekty specjalne w filmach,
- gry komputerowe.

Początki grafiki komputerowej sięgają lat 50. XX wieku, jednak ze względu na duże koszty komputerów i urządzeń graficznych aż do lat 80. grafika komputerowa była wąską specjalizacją, a na jej zastosowania praktyczne mogły pozwolić sobie ośrodki badawcze, duże firmy oraz instytucje rządowe. Dopiero gdy w latach osiemdziesiątych rozpowszechniły się komputery osobiste, grafika komputerowa stała się czymś powszechnym.

Klasyfikacja grafiki prowadzona jest według dwóch kryteriów: techniki tworzenia obrazu i liczby wymiarów. Z punktu widzenia techniki tworzenia rozróżniamy grafikę wektorową i rastrową. Ze względu na wymiar wyróżniamy grafikę dwuwymiarową (2D) oraz trójwymiarową (3D). Natomiast ze względu na metodę pracy i czas grafikę dzielimy na nieinterakcyjną, interakcyjną i czasu rzeczywistego.

5.3.1. GRAFIKA RASTROWA

Jest to sposób tworzenia grafiki komputerowej, w której obraz zdefiniowany jest jako zbiór bardzo małych niezależnych od siebie punktów (pikseli) tej samej wielkości ułożonych równomiernie w wierszach i kolumnach. Rastrem nazywa się siatkę tych punktów. Obrazy utworzone w ten sposób nazywa się bitmapami. Nazwa jest pozostałością historyczną z czasów monitorów monochromatycznych, w których punkt świecił lub nie. Zatem jego stan mógł być przedstawiony za pomocą jednego bitu. Typowe formaty to BMP, TIFF, GIF i JPEG.

5.3.2. GRAFIKA WEKTOROWA

W tej metodzie tworzenia grafiki komputerowej obraz powstaje na podstawie matematycznego opisu określającego pozycję, długość i kierunek prowadzonych linii. Na przykład dla okręgu pamiętane są współrzędne środka i promień, dla odcinka współrzędne końców. Do tworzenia obrazu stosuje się między innymi krzywe Béziera. Obraz jest uporządkowanym zbiorem wektorów. Zaletą tego sposobu opisu obrazu jest skalowalność. Chociaż do projektowania grafiki używa się opisu wektorowego, to końcowy wynik przedstawia się w postaci rastrowej. Do lat 80. XX w. wykorzystywane były monitory CRT, na których obraz powstawał w oparciu na grafice wektorowej, w sposób analogiczny jak w oscyloskopie. Przykładowy format to CDR z programu graficznego CorelDraw.

5.3.3. GRAFIKA TEKSTOWA

Obecnie prawie niespotykany sposób wyświetlania lub drukowania obrazu za pomocą znaków ze zbioru znaków ASCII, w szczególności z wykorzystaniem znaków semigraficznych. Nie można za ich pomocą oddać drobnych detali, można jednak uzyskać interesującą oprawę graficzną tekstu. Przykład grafiki tekstowej zawiera rys. 5.1. Obecnie w Internecie można spotkać elementy takiej twórczości pod nazwą ASCII Art lub Text Art. Wymagana jest stała wysokość i szerokość czcionki.

CYBER-72



Rysunek 5.1. Grafika znakowa.

5.3.4. GRAFIKA DWUWYMIAROWA

W tym rodzaju grafiki wszystkie obiekty są płaskie i w szczególności obrazy rastrowe mieszczą się w tej kategorii. W skrócie nazywamy ją grafiką 2D. Grafikę taką wykorzystuje się wszędzie tam, gdzie istnieje potrzeba stosowania obrazów płaskich, na przykład w kartografii, kreślarstwie, reklamie, etykietach i tym podobnych, w których obraz dwuwymiarowy lepiej oddaje rzeczywistość lub jest bardziej użyteczny.

Grafika 2D swoje pierwsze zastosowania znalazła w dziedzinie komputerów w latach 50. XX w. Najpierw była to grafika wektorowa, która w miarę rozwoju nowych technologii kart graficznych i monitorów została wyparta przez grafikę rastrową.

W grafice dwuwymiarowej stosuje się techniki wektorowe, rastrowe oraz znaki tekstu, czyli czcionkę. Obrazy mogą być modyfikowane przez takie operacje matematyczno-geometryczne, jak skalowanie, obroty i przesunięcia.

Interesującą możliwością w grafice dwuwymiarowej jest użycie warstw. Grafika zawierająca warstwy jest czasem nazywana grafiką dwuipółwymiarową. Ilustracją tej techniki jest obraz, którego elementy zapisano na przezroczystych foliach, których ułożenie jedna na drugiej daje kompletny obraz, a usuwając niektóre z nich, uzyskujemy obraz pozbawiony wybranych szczegółów. Takie rozwiązanie jest przydatne na przykład w systemach informacji przestrzennej, gdzie możemy składać obraz z różnych elementów infrastruktury. Jedna warstwa to rzeźba terenu, inna to sieć drogowa, kolejne to sieć gazowa, sieć elektryczna, sieć teletechniczna, zabudowa itd. W zależności od potrzeb możemy wizualizować odpowiednie warstwy.

5.3.5. GRAFIKA TRÓJWYMIAROWA

Zadaniem tej dziedziny grafiki komputerowej jest przedstawienie trójwymiarowej rzeczywistości na płaskim, dwuwymiarowym obrazie. W skrócie określana jako grafika 3D. W porównaniu z grafiką 2D geometria grafiki 3D jest o wiele bardziej złożona. Możemy tu skorzystać z kilku sposobów.

Siatka wielokątów. Powierzchnia ograniczająca obiekt składa się z siatki wielokątów o wspólnych wierzchołkach i krawędziach. Najczęściej są to trójkąty lub czworokąty. Im siatka jest gęstsza, tym obiekt może być lepiej odwzorowany i na dodatek może być bardziej skomplikowany.

Voxele (woksele). Jest to naturalne uogólnienie siatki dwuwymiarowej (rastra) na przestrzeń trójwymiarową. Obraz budowany jest z elementarnych sześcianów, trójwymiarowych odpowiedników pikseli. Taka reprezentacja stosowana jest na przykład w diagnostyce medycznej, w której z szeregu płaskich przekrojów buduje się trójwymiarowy model.

Opis matematyczny. Ten sposób jest uogólnieniem dwuwymiarowej grafiki wektorowej na przestrzeń trójwymiarową. Elementy obrazu opisane są równaniami z tą różnicą, że mamy tu także kule, płaszczyzny, a krzywe Béziera zastąpione są powierzchniami, np. Béziera czy Hermite'a.

Dane do grafiki 3D mogą zostać pobrane ze świata rzeczywistego z wykorzystaniem zdjęć satelitarnych, skanerów trójwymiarowych, zdjęć stereoskopowych lub tomografów komputerowych. Elementy obrazu mogą też zostać przygotowane przez człowieka w procesie modelowania.

Kolejny problem grafiki trójwymiarowej to wizualizacja. Napotykamy tu problemy związane z perspektywą, oświetleniem i przedstawianiem powierzchni widocznych i zasłoniętych. Z takimi samymi problemami mamy do czynienia np. w fotografii czy malarstwie. W grafice 3D wprowadza się pojęcie wirtualnej kamery, a z jej położeniem wiąże się punkt widzenia i wizualizację perspektywy, powierzchni, światłocieni.

W grafice trójwymiarowej na uwagę zasługuje realizm. Dla jego uzyskania modeluje się oświetlenie, fakturę powierzchni, cienie rzucane przez obiekty, odbicia zwierciadlane, załamania i rozproszenie światła i tym podobne elementy obrazu. Niestety pełny realizm jest kosztowny obliczeniowo, dlatego w niektórych zastosowaniach, na przykład w grach komputerowych stosuje się metody przybliżone.

5.3.6. GRAFIKA NIEINTERAKCYJNA

W tym typie grafiki program tworzy obraz na podstawie przygotowanych wcześniej danych. Takie podejście jest bardzo użyteczne w systemach do modelowania oraz wszędzie tam, gdzie potrzebna jest powtarzalność tworzonego obrazu. Rozwiązanie to może być przydatne także tam, gdzie dane do tworzenia obrazu pobierane są z urządzeń pomiarowych.

5.3.7. GRAFIKA INTERAKCYJNA

W tym rozwiązaniu program tworzy obraz zgodnie z działaniami użytkownika i zmienia wygląd ekranu tak, że użytkownik od razu widzi skutki swoich poleceń. Niezwykle ważne jest, aby czas odświeżania obrazu nie był zbyt długi, czego konsekwencją jest stosowanie uproszczonych metod rysowania obiektów. Ta technika jest powszechnie stosowana we wszystkich ogólnie znanych programach do tworzenia grafiki.

5.3.8. GRAFIKA CZASU RZECZYWISTEGO

Technika ta wymaga szybkiego tworzenia obrazu i jego wizualizacji; obraz musi być regenerowany wiele razy na sekundę. Grafikę czasu rzeczywistego stosuje się przede wszystkim w różnego rodzaju symulacjach oraz grach komputerowych.

5.4. PROGRAMY DO TWORZENIA OBRAZU

Do tworzenia grafiki rastrowej (bitmatowej) używa się **programów malarskich**, które umożliwiają kolorowanie poszczególnych punktów obrazu poprzez definiowanie ich barw. Odbywa się to na ogół poprzez barwienie pikseli tworzących mapę bitową obrazu na monitorze. Za pomocą programu malarskiego można w prosty sposób osiągnąć ciekawe efekty, ale ich modyfikacja jest trudna, gdyż nie można wydzielić z nich pojedynczych obiektów. Przykładami programów malarskich są: **MS Paint, Adobe PhotoShop, Micrografix Picture Publisher**.

Do tworzenia grafiki wektorowej (obiektowej) stosujemy **programy rysunkowe**, które umożliwiają tworzenie obrazów kreskowych. Elementy składowe rysunku przechowywane są w postaci wzorów matematycznych, a to umożliwia ich skalowanie bez wprowadzania zniekształceń, a także ich prezentację na urządzeniach drukujących (np. drukarki, naświetlarki dla matryc drukarskich) z maksymalną rozdzielczością. Punkty definiujące

krzywe, z których składa się obraz nazywamy uchwytami. Na przykład dla odcinka są to dwa punkty, dla krzywych Béziera wystarczą, oprócz końców, dwa dowolne punkty. W programach grafiki wektorowej punkty te są zazwyczaj oznaczane małymi kwadracikami. Przeciągając uchwyty, można zmieniać położenie obiektu, jego kształt lub skalę. Jako przykłady programów rysunkowych można wymienić: CorelDraw, WordPerfect Graphics, Macromedia Freehand, Adobe Illustrator.

Grafika wektorowa ma przewagę nad grafiką rastrową, nie tylko ze względu na skalowalność, ale też z tego powodu, że istnieje dokładna informacja o elementach, z których składa się obraz. Dzięki temu w programach rysunkowych istnieje łatwość korekty całych obiektów. Natomiast w programach malarskich możemy modyfikować pojedyncze piksele, a tylko czasami ich grupy.

5.5. PROGRAMY DLA INŻYNIERÓW

W projektowaniu wspomaganym komputerowo (CAD/CAM) komputer wykorzystywany jest do projektowania przemysłowego i wykonywania rysunku technicznego. Oprogramowanie klasy CAD (ang. Computer-Aided Design) może być stosowane zarówno przez inżynierów, wykonujących projekt techniczny, jak i przez artystów zajmujących się wizualizacją, czyli projektujących wygląd zewnętrzny. Programy klasy CAD wykorzystywane są w wielu dziedzinach projektowania, takich jak: architektura, inżynieria cywilna, inżynieria elektryczna, mechanika precyzyjna czy dekoracja wnętrz. Najbardziej znanym programem z tej dziedziny jest *AutoCAD*.

Aplikacje CAD wykonują bardzo wielką liczbę obliczeń i wymagają precyzyjnej grafiki, a zatem potrzebują szybkich procesorów i monitorów o dużej rozdzielczości. Wyniki projektowania CAD mogą być bezpośrednio przekazane do sterowanego komputerowo urządzenia produkcyjnego (CAM, ang. Computer-Aided Manufacturing), co umożliwi rozpoczęcie procesu produkcji prawie natychmiast po zakończeniu projektowania. Klasyfikacyjnym przykładem są obrabiarki sterowane numerycznie.

5.6. KOMPRESJA DANYCH

Podstawowym celem kompresji danych jest takie przekształcenie informacji, aby zajmowała mniej miejsca. Matematycznie uzyskuje się to poprzez zmniejszenie redundancji. Pliki skompresowane mogą zostać odczytane dopiero po wykonaniu operacji odwrotnej, czyli dekompresji.

Kompresję dzielimy na dwie grupy: **stratną** i **bezstratną**. Kompresja bezstratna umożliwia z postaci spakowanej odzyskanie informacji w postaci oryginalnej. Ma ona znaczenie między innymi w przesyłaniu danych i przechowywaniu dokumentów. W kompresji stratnej odzyskanie z postaci spakowanej oryginalnej informacji nie jest możliwe, jednak podstawowe informacje i istotne właściwości obiektu zostają zachowane.

Algorytmy kompresji opierają się na modelach prawdopodobieństwa. Dzielimy je na algorytmy zastosowania ogólnego oraz algorytmy dopasowane do określonego typu da-

nych. Jeden z najbardziej znanych i popularnych algorytmów kompresji bezstratnych to kod Huffmana opracowany przez Davida A. Huffmana na początku lat 50. XX w. Algorytmy kompresji stratne stosuje się na przykład w kompresji obrazu i dźwięku, co jest możliwe ze względu na niedoskonałość ludzkich zmysłów. Przykładem w grafice jest format JPEG, w technice wideo (kodowanie filmów) – MPEG, w muzyce – MP3.

5.7. DŹWIĘK W KOMPUTERZE

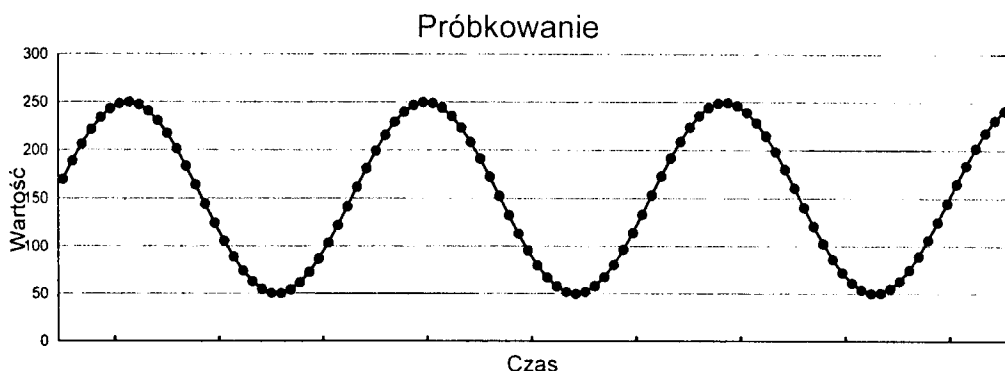
Komputer pracuje w sposób nieciągły, dyskretny, natomiast dźwięk w swojej naturze jest zjawiskiem ciągłym. Stąd konieczność zamiany dźwięku na postać nieciągłą poprzez próbkowanie, czyli zapisywanie parametrów dźwięku w określonych momentach czasowych. Tu podobnie jak przy zapisie obrazu mamy różne formaty.

Na początek trochę informacji o formacie WAV zapewniającym jakość płyty kompaktowej, a opracowanym dla systemu operacyjnego Microsoft Windows przy współpracy z firmą IBM. Ze względu na wysoką częstotliwość próbkowania 44.1 kHz, zapis na 16 bitach i w dodatku stereofonicznie na zapisanie 1 sekundy dźwięku potrzeba 172 kB. Stąd zapis kilkuminutowego utworu zajmuje dziesiątki megabajtów.

Chcąc oszczędzić miejsce, musimy zrezygnować z jakości dźwięku. Możemy to zrobić na kilka sposobów. Na przykład: do zapisu używać tylko 8 bitów, zrezygnować z dźwięku stereo, zmniejszyć częstotliwość próbkowania do 22 kHz lub 8 kHz.

Przy minimalnych ustawieniach, 8 kHz, 8 bitów, dźwięk mono, 1 sekunda dźwięku zajmuje 8 kB.

Rysunek 5.2. przedstawia poglądowo ideę próbkowania. Krzywa reprezentuje jakiś dźwięk, a kropki wartości pobrane przez urządzenie cyfrowe rejestrujące ten dźwięk. Czas to 100 ms, do zapisu wartości używamy 8 bitów, próbkowanie jest co 1 ms. Ciąg liczb poniżej rysunku zawiera pierwszych 10 odczytanych wartości, a poniżej ich postać binarną.



Rysunek 5.2. Próbkowanie dźwięku.

170	189	206	222	234	243	249	250	247	241	...
10101010	10111101	11001110	11011110	11101010	11110011	11111001	11111010	11110111	11110001	...

Pasek poniżej ilustruje zapis binarnej reprezentacji powyższych liczb w postaci ścieżki np. na dysku twardym, czarne pola to jedynki, białe to zera.



Jednak na potrzeby wymiany plików muzycznych potrzebny był format, który zapewniając dobrą jakość dźwięku, będzie potrzebował mniej miejsca do zapisu. Krótko omówię dwa najpopularniejsze formaty stratnej kompresji dźwięku.

MP3 (ang. *MPEG-1/2 Audio Layer-3*) – ten format stratnej kompresji dźwięku został opracowany w 1991 r. w Instytucie Fraunhofera. Format został zaprojektowany tak, aby umożliwić poprawne działanie mimo przekłamań bądź utraty części danych. Podczas projektowania starano się, aby format mimo kompresji dobrze oddawał brzmienie ludzkiego głosu. Zakodowany dźwięk może być stereofoniczny lub monofoniczny. Przy dźwięku stereofonicznym możemy kodować każdy kanał oddzielnie lub zastosować metodę różnicową, kodując sumę monofoniczną obu kanałów oraz sygnał różnicy tychże.

Ogg to format darmowego kontenera strumieni danych. Format ten jest wspierany przez fundację Xiph.org w związku z rozwojem wolnego oprogramowania do kodowania i dekodowania multimediiów. Zarazem jest to nazwa rodziny kodeków. Kontener może zawierać wiele strumieni dźwięku, obrazu oraz napisów.

W ramach prac nad kontenerem Ogg powstał format stratnej kompresji dźwięku **Vorbis**. Używany w połączeniu z kontenerem Ogg nosi nazwę **Ogg Vorbis**. Format ten potrafi obsłużyć do 255 kanałów i 16-bitowy dźwięk w zakresie 6–48 kHz.

Pliki formatu Ogg Vorbis zapewniają wyższą jakość dźwięku przy wysokiej kompresji niż pliki formatu MP3. Format Ogg Vorbis nie jest opatentowany i pozostaje bezpłatny zarówno do celów prywatnych, jak i komercyjnych, w przeciwieństwie do formatu MP3. Jedyną wadą jest wyższe zapotrzebowanie na moc obliczeniową niż przy formacie MP3.

6. EDYCJA TEKSTU

Zacniemy od krótkich definicji:

- [1999a] EDYTOR TEKSTÓW – program przekształcający komputer w narzędzie do tworzenia, redagowania, korekty, formatowania i drukowania dokumentów. Przykładami takiego programu są Lotus Word Pro, Microsoft Word i WordPerfect. Programy służące do przetwarzania tekstów znajdują się na pierwszym miejscu listy najlepiej sprzedającego się oprogramowania z prostego powodu: spośród wszystkich programów komputerowych są najpowszechniej używane. Współczesne edytory tekstów mają wiele cech czyniących pracę pisarzy i redaktorów prostszą. Funkcja „znajdź i zamień” ułatwia poprawianie dokumentu, podczas gdy słownik ortograficzny oraz słownik wyrazów bliskoznacznych zapewniają bezbłędną, czytelną końcową formę dokumentu. Różne narzędzia do formatowania umożliwiają nadanie tekstowi atrakcyjnego wyglądu graficznego. Wraz ze wzrostem popularności systemów wykorzystujących graficzny interfejs użytkownika (GUI) – jak Macintosh i zgodne z IBM PC komputery z systemem Microsoft Windows 95 – działające w nich programy zostały wyposażone w możliwość wyświetlania na ekranie różnych czcionek o różnych rozmiarach. Za pomocą współczesnych edytorów tekstu można bez problemu wykonywać wszystkie proste zadania składu komputerowego, np. produkcję broszur reklamowych.
- [2000a] EDYTOR TEKSTU – program do tworzenia i edycji tekstu. Edytor tekstu zastępuje użytkownikowi komputera kartkę papieru, długopis, maszynę do pisania, gumkę, zestaw do kolorowania, a najczęściej także słowniki ortograficzny i wyrazów bliskoznacznych. Prostymi edytorami tekstu, które użytkownik znajdzie w systemie Windows, są programy, Notatnik i WordPad. Najbardziej znanym na świecie zaawansowanym edytorem tekstu jest Word firmy Microsoft.
- [1999a] SKŁAD KOMPUTEROWY (DTP) – użycie komputera osobistego jako podstawy niedrogo systemu przygotowywania tekstu do druku. W DTP często łączy się tekst z grafiką i wynik drukuje na drukarce laserowej o dużej rozdzielczości lub na naswietlarce. Oprogramowanie DTP pozwala jednej osobie na przygotowanie tekstu i grafiki do

druku za pomocą komputera osobistego, dzięki czemu można zmniejszyć koszty przygotowania publikacji nawet o 75%.

- [2000a] DTP – profesjonalne łamanie tekstu, wstawianie grafiki i drukowanie przy użyciu komputera. Programy do DTP (np. QuarkXPress, czy Ventura Publisher) najpierw importują tekst i ilustracje, a następnie operator DTP nadaje tym materiałom formę, którą później można oglądać w książkach czy gazetach – zawierającą szpalty, podpisy do ilustracji, tytuły itd. Od ang. Desktop Publishing – w wolnym tłumaczeniu publikacje tworzone na pulpicie komputera.

6.1. O HISTORII REDAGOWANIA TEKSTÓW

Komputery osobiste używane są jako uniwersalna maszyna do pisania, a przygotowana za ich pomocą publikacja nie ustępuje jakością profesjonalnemu składowi tekstu.

Programy przeznaczone do pisania i redagowania tekstów znane są pod nazwą edytorów tekstowych (*text editors*) lub programów przetwarzania tekstów (*word processors, text processors*). Używana powszechnie nazwa edytor tekstu w zasadzie jest zarezerwowana dla oprogramowania przeznaczonego do pisania i poprawiania wersji źródłowych programów w różnorodnych językach programowania.

Jednym z pierwszych programów redagowania tekstów był Wordstar firmy MicroPro. Jego autorem jest Robert Barnaby, a powstał w 1978 r. z przeznaczeniem na komputery pracujące z systemem operacyjnym CP/M. Najpierw określano go Wordmaster, a w 1979 r. nadano mu obecną nazwę. Odpowiednie wersje programu Wordstar zostały przygotowane na wiele typów komputerów i systemów operacyjnych, m.in. na popularne niegdyś w Polsce mikrokomputery Amstrad i IBM PC XT z systemem DOS. Edytor cechowała duża szybkość działania, małe zapotrzebowanie na pamięć komputera, tylko 64 KB, podczas gdy obecnie procesory tekstu wymagają ok. 32 MB pamięci. Program ten stał się wzorem do naśladowania, a zarazem wprowadził do programów redagowania tekstów pewne standardy. W Polsce pod koniec lat 80. XX wieku analogicznymi programami rodzimej produkcji były PL-Tekst CSK (firmy CSK), CX-Text (firmy Computex) i Wedstar (firmy Mikrokomputery).

Nowoczesne programy redagowania cechuje wysoka jakość przedstawienia końcowej wersji redagowanego tekstu na ekranie monitora. Podczas redakcji dokumentu wszystkie nanoszone poprawki, zarówno w zakresie tekstu, jak i jego układu, znajdują odbicie na ekranie monitora. Obecnie zasadą w programach przetwarzania tekstów jest „masz to, co widzisz”, a więc zapewnienie, że tekst na ekranie odpowiada dokładnie tekstowi wydrukowanemu. Niestety, ta zasada jest tylko w części realizowana, gdyż możliwości drukarki różnią się od możliwości monitora.

6.2. OPERACJE NA PLIKACH

Do podstawowych operacji na plikach zawierających dokumenty tekstowe zaliczamy: otwarcie dokumentu i nadanie nazwy, zmianę nazwy, łączenie plików, skasowanie, wydrukowanie pliku oraz wyświetlenie katalogu dokumentów. Zauważmy, że te dzia-

łania można w zasadzie przeprowadzić, korzystając z możliwości wykonywania operacji na plikach wbudowanych w system operacyjny. Najbardziej istotną operacją jest łączenie plików. Nie może być ona w sposób interesujący dla autora tekstu realizowana przez system operacyjny, gdyż pozwala on tylko na dołączanie pliku na końcu innego pliku. Natomiast w programie przetwarzania tekstu użytkownik może włączyć fragment tekstu zapisany w innym pliku w dowolne miejsce redagowanego tekstu. Możemy budować dokument przez wstawianie tekstu z wielu plików, włączając cały plik lub jego część.

6.3. EDYCJA TEKSTU

Edycja, czyli pisanie tekstu i wprowadzanie poprawek, polega na wpisywaniu bądź kasowaniu w dowolnym miejscu nowych znaków, słów czy dowolnych fragmentów oraz przemieszczaniu lub usuwaniu wybranych fragmentów (bloków) tekstu, a także na przeglądaniu całego dokumentu.

Wprowadzamy znaki tekstu z klawiatury, tak jak na maszynie do pisania, i są one natychmiast wyświetlane na ekranie w miejscu wskazywanym przez kursor. Do korekty tekstów służą klawisze [Ins], [Del] i [BackSpace] (←).

Klawisz [Ins] (od ang. *insert* = wstawiać) działa jako przełącznik trybu wstawiania. Wprowadzane z klawiatury znaki są umieszczane w miejscu wskazywanym przez kursor, a jeśli jesteśmy w środku tekstu, to jest on przesuwany w prawo.

Klawisz [Del] (od ang. *delete* = usuwać) powoduje usunięcie znaku wskazywanego przez kursor, a klawisz [BackSpace] (odpowiednik cofacza na maszynie do pisania) usuwa znak na lewo od kursora. W efekcie tekst z prawej strony zostaje przesunięty w lewo.

Współczesne edytory tekstów wyświetlają na ekranie nazwę redagowanego dokumentu oraz parametry położenia kursora: aktualną stronę, wiersz i kolumnę lub położenie względne na stronie w centymetrach. Ponadto, wyświetlana jest informacja o trybie pracy (stanie klawisza [Ins]) i o położeniu marginesów. W górnej części ekranu wyświetlone jest menu, a poniżej menu mogą być wyświetlane paski narzędzi. Ich wygląd może zmieniać użytkownik. Można także wyświetlić wiersz formatowy (linijkę), obrazujący ustawienia lewego i prawego marginesu oraz tabulatorów.

Zmiana położenia kursora na ekranie odbywa się poprzez klawisze sterujące kursorem oraz rozkazami zbudowanymi z kombinacji klawisza [Ctrl] i klawiszy znakowych. Klawisze [Home] i [End] służą na ogół do umieszczenia kursora na początku lub końcu wiersza. Klawisze [PgUp] i [PgDn] służą do przeglądania dokumentu po jednej stronie ekranu, czasami po jednej stronie logicznej dokumentu, odpowiednio w kierunku początku i końca dokumentu. Przy użyciu pojedynczych klawiszy kursorów można przeglądać tekst po jednym wierszu do przodu lub do tyłu, po jednym znaku w prawo lub w lewo. Kursor można też przesuwać o słowo w lewo lub w prawo, jednym skokiem do lewego lub prawego marginesu, a także do początku lub do końca dokumentu.

Podczas edycji użytkownik może usunąć znaki, słowa lub wiersze. Usuwany fragment tekstu jest zapamiętywany w wydzielonym obszarze, zwanym buforem, dzięki temu istnieje możliwość jego odzyskania.

Dowolny fragment tekstu może być wyróżniony jako blok i jest wyświetlony w innym kolorze niż reszta tekstu. Odpowiednie rozkazy pozwalają usunąć, przenieść lub skopiować ten blok w inne wskazane miejsce dokumentu lub zapisać pod wybraną nazwą jako plik na dysku.

Bardzo istotną cechą współczesnych edytorów tekstu jest możliwość pracy w trybie kolumnowym, który polega na podzieleniu strony na równoległe strumienie tekstu oddzielone od siebie odstępem lub pionową linią.

Redaktor tekstu umożliwia użytkownikowi wyszukiwanie w tekście podanych ciągów znaków i ewentualne zastąpienie ich innymi. Możemy przy wyszukiwaniu wymagać, aby były rozróżniane duże i małe litery. Użytkownik może zażądać, aby operacja została wykonana automatycznie w całym dokumencie lub aby program zatrzymywał się po napotkaniu wybranego ciągu znakowego i czekał na potwierdzenie. Niedostatkami programu redakcyjnego jest brak możliwości wyszukania i wymiany niektórych kodów sterujących.

6.4. FORMATOWANIE NA EKRANIE

Celem operacji formatowania ekranowego jest przygotowanie szaty graficznej dla wydrukowanego dokumentu. Użytkownik ustawia i zmienia marginesy (lewy, prawy, dolny, górny), sposób wyrównania tekstu, możliwość dzielenia przenoszonych wyrazów, środkowanie tekstu, tzn. automatyczne umieszczenie na środku wiersza lub strony. Kształt dokumentu widzimy na ekranie i możemy go wielokrotnie zmieniać.

Automatyczną funkcją formatowania tekstu jest przenoszenie wyrazu do następnego wiersza po przekroczeniu wybranej szerokości szpalty. Klawisz [Enter/Return] stosujemy tylko w celu zamierzonego przejścia do następnego wiersza. Na przykład jeśli aktualny wiersz ma być krótszy niż szerokość szpalty, a więc jeśli jest to koniec akapitu, tytuł itp. kod sterujący odpowiadający użyciu klawisza [Enter] jest zapamiętywany w tekście. Tekst zawarty pomiędzy kolejnymi dwoma takimi kodami sterującymi nazywamy akapitem (ang. *paragraph*). Akapit jest jedną z podstawowych jednostek formatowania.

Następną ważną opcją jest justowanie (ang. *justification*), czyli wyrównywanie tekstu w poszczególnych wierszach do prawego marginesu. Odbywa się to na ogół przez równomierne wstawienie spacji między wyrazy w wierszu, w taki sposób, aby ostatni wyraz kończył się dokładnie na prawym marginesie. Jeśli nie włączy się tej opcji, wówczas prawy margines tekstu jest nierówny, podobnie jak podczas pisania na maszynie. Tekst wyrównany do prawego marginesu upodabnia się do tekstu składanego w drukarni, do tekstu w książce, a tekst z niewyrównanym marginesem przypomina pisanie na maszynie. Niektórzy uważają, że w korespondencji prywatnej bardziej elegancko jest pozostawić nierówny margines.

Współczesne programy redagowania tekstów umożliwiają justowanie lewostronne, prawostronne i obustronne. Justowanie lewostronne polega na wyrównaniu tekstu na lewym marginesie z pozostawieniem poszarpanego prawego marginesu. Justowanie obustronne polega na wyrównaniu tekstu do lewego i prawego marginesu. Justowanie prawostronne oznacza, że tekst jest wyrównany do prawego marginesu, lecz ma poszarpany lewy margines. Po włączeniu tej opcji kursor ustawia się na prawym marginesie, a każdy nowo wpisany znak przesuwają poprzednio wpisany tekst w lewo. Jest to przydatne na przykład przy wstawianiu daty w prawym górnym rogu dokumentu.

O wyglądzie dokumentu decyduje również wybrany odstęp międzywierszowy, czyli interlinia. Często stosujemy na początku wprowadzania tekstu pojedynczy odstęp międzywierszowy, który zmieniamy pod koniec redakcji tekstu.

Procesor tekstu dokonuje automatycznego podziału na strony i podział ten możemy oglądać na ekranie. Można go też zmienić.

Podczas zmian marginesów, odstępu między wierszami, justyfikacji, wprowadzania korekt następuje automatyczne reformatowanie tekstu. Wówczas spacje dopisane automatycznie są usuwane, a dodawane są inne. Natomiast spacje wpisane z klawiatury przez użytkownika są pozostawiane i tylko przez niego mogą być usunięte, podobnie tak jak inne znaki tekstu. Wcięcie utworzone przez spacje na początku akapitu nie jest podczas formatowania zmieniane. Jeśli chcemy, aby w określonym miejscu była dokładnie jedna spacja, to wpisujemy ją przy użyciu odpowiedniego rozkazu (tzw. twarda spacja). Stosujemy to na przykład, gdy chcemy, aby na końcu wiersza jednostka miary nie została oddzielona od liczby i nie została samotna litera.

6.5. FORMATOWANIE WYDRUKU

W chwili wywołania rozkazu wydruku następuje automatyczne formatowanie wydruku, które polega na umieszczeniu w tekście znaków sterujących drukarką w celu uzyskania takich efektów jak na ekranie.

Obecnie drukowanie odbywa się w trybie graficznym i przygotowanie wydruku polega na przełożeniu wyglądu ekranu na obraz drukowany przez drukarkę.

W momencie rozpoczynania wydruku z procesora tekstu można wybrać urządzenie docelowe (np. drukarkę lub plik), zadać numer strony początkowej i końcowej, zdecydować, czy wydruk ma być zatrzymany po każdej stronie dla zmiany kartki w drukarce.

6.6. ZNAKI NA EKRANIE I NA DRUKARCE

W odróżnieniu od maszyny do pisania komputer nie tylko ułatwia korektę tekstów, ale umożliwia rozszerzenia zestawu stosowanych znaków o znaki różnych alfabetów lub znaki graficzne. We wcześniejszych wersjach redaktorów tekstu zestaw znaków miał przypisane kody ASCII od 0 do 127, a rozszerzony zestaw – kody ASCII 0 do 255. Dodatkowe znaki wpisywane są przez wciśnięcie klawisza [Alt] i wybranie odpowiedniego kodu ASCII na klawiszach numerycznych po prawej stronie klawiatury. W systemie MS Windows 95 został wprowadzony nowy sposób numerowania znaków i trzeba z klawiszem [Alt] wcisnąć cztery cyfry strony kodowej 1250.

Jakość tekstu uzyskiwana na monitorze jest obecnie bardzo wysoka. Jakość druku zależy natomiast tylko od trybu pracy drukarki. Drukarki mozaikowe mają rozdzielczość rzędu 240 punktów/cal, a drukarki atramentowe i laserowe 300–1200 punktów/cal. Dla porównania foteskład ma rozdzielczość 1200 punktów/cal.

Jedynym problemem jest interpretacja przez drukarkę znaków strony kodowej. Może to powodować błędy w tekście, szczególnie w zakresie znaków diakrytycznych.

6.7. ROZSZERZENIA I UDOGODNIENIA

We wszystkich programach redagowania tekstów istnieje możliwość jednoczesnej pracy z wieloma dokumentami umieszczonymi w różnych oknach ekranu, przywrócenia skasowanego fragmentu tekstu (ang. *undelete*), umieszczenia w stopce przypisów, które automatycznie są numerowane i przenoszone na odpowiednią stronę podczas redagowania tekstu, automatycznego sporządzania spisów treści na podstawie zaznaczonych w tekście tytułów rozdziałów i sporządzania indeksów (skorowidzów) na podstawie zaznaczonych w tekście słów, sortowania umieszczonych w tekście zestawień, operacji sumowań w tabelach, rysowania linii pionowych i poziomych, a także wstawiania grafiki.

Współczesne edytory umożliwiają tworzenie makrorozkazów, które są wywoływane podczas pracy edytora lub połączone z klawiszami funkcyjnymi. Daje to dużą elastyczność w dostosowywaniu programu do konkretnych celów. Mamy także możliwość tworzenia szablonów, np. do konstruowania formularzy, papierów firmowych, do pisania listów itp. Redaktory tekstu umożliwiają ponadto: łączenie tekstów z kolejnymi elementami listy, np. przy drukowaniu korespondencji do adresatów z listy na podstawie jednego standardowego tekstu (korespondencja seryjna), sprawdzanie poprawności ortograficznej tekstów, podpowiadanie wyrazów bliskoznacznych i inne. Lista adresatów może zostać przygotowana w programie redagowania tekstów lub być uzyskana jako raport z bazy danych, np. jako spis osób spełniających określone kryteria.

Podczas weryfikacji poprawności ortograficznej program porównuje tekst z zapisanym na dysku słownikiem. Po stwierdzeniu błędu lub po napotkaniu słowa, które nie występuje w słowniku, słowo w tekście dokumentu jest zaznaczone i może być poprawione lub zastąpione przez inne. Współczesne edytory wyświetlają listę wyrazów o zbliżonej pisowni lub wymowie, a wybrane słowo jest automatycznie wstawiane do tekstu. Użytkownik ma możliwość tworzenia własnego pomocniczego słownika zawierającego nowe słowa, a także może założyć pomocnicze słowniki ze specjalistyczną terminologią. Można też połączyć z edytorem tekstów słownik języka obcego, co umożliwia pomoc w tłumaczeniu.

W edytorach tekstów możliwe jest przeprowadzenie analizy leksykograficznej, na przykład policzenie liczby słów i liter w tekście, liczby zdań, wierszy i akapitów. Można też uzyskać informacje o liczbie liter w wyrazie i średniej liczbie wyrazów w zdaniu.

6.8. WYMIANA TEKSTÓW MIĘDZY PROGRAMAMI

Programy redakcji tekstu umożliwiają odczytanie tekstów napisanych w innych programach, pod warunkiem że nie zawierają zakodowanych informacji formatowych.

Obecnie wszystkie edytory tekstu zapisują wraz z tekstem własne informacje o formatowaniu dokumentu. Z tego powodu bierze się konieczność dołączania zestawu konwerterów pomiędzy edytorami tekstu różnych producentów, a często także pomiędzy różnymi wersjami tego samego edytora.

Dla ułatwienia wymiany tekstów pomiędzy różnymi edytorami ustalono standard zapisu w postaci uproszczonej, tzw. Rich Text Format (RTF).

6.9. SKŁAD TEKSTU – DTP

Postęp w rozwoju sprzętu komputerowego klasy PC, a w szczególności oprogramowania, umożliwił skład tekstu i druk publikacji w niewielkich nakładach. Zastosowanie komputerów do małej poligrafii określane jest w języku angielskim jako *desktop publishing*. Jak wynika z przytoczonych na wstępie definicji, programy z tej dziedziny powinny umożliwiać automatycznie zaawansowane formatowanie wydruku zgodnie z zadaniem wzorcem, umożliwiając łamanie w kolumnach, łączenie tekstu z grafiką oraz generowanie wydruku o wysokiej jakości. Użytkownik obserwuje, jak formatowany jest dokument na ekranie w trybie graficznym, a więc w sposób interaktywny. Są jednak programy, w których dokument przygotowany jest w trybie tekstowym, a formatowanie następuje podczas przygotowania wydruku – taki tryb postępowania nazywamy wsadowym.

Do dziś bardzo popularnym wsadowym programem składania tekstów (ang. *typesetting*) jest PC TeX (Personal TeX). Użytkownik przygotowuje tekst do wydruku za pomocą dowolnego edytora, wpisując rozkazy formatowania pomiędzy swój tekst. Zanim wydrukujemy, mamy możliwość sprawdzenia na ekranie dokumentu w formie zbliżonej do tej, w jakiej będzie drukowany. Teksty i wzory złożone przy użyciu tego programu zyskały szeroką akceptację matematyków. Przede wszystkim dlatego, że autorem koncepcji zastosowanej w tych programach jest matematyk Donald E. Knuth. Napisał on na ten temat książkę *The TEXbook* opublikowaną przez Addison-Wesley Publishing Corp. w 1984 r. Języki umożliwiające formatowanie dokumentów w sposób wsadowy zwane są w skrócie językami PDL od ang. Page Description Language. Obecnie najbardziej popularny z nich to PostScript opracowany przez firmę Adobe Systems. Wiele współczesnych drukarek laserowych jest dostosowanych do interpretacji rozkazów tego języka, np. firmy Hewlett-Packard. Język ten jest również stosowany w pakiecie PageMaker firmy Aldus. Alternatywny język DDL (Document Description Language), opracowany przez Imagen, stosowany jest m.in. w pakiecie Ventura Software rozprawdzanym kiedyś przez Xerox. Na zakończenie tego rozdziału przedstawię kilka rad dotyczących tworzenia publikacji, pochodzących z książki opublikowanej w 1994 r. (por. Źródła [1994b]).

SZEŚĆ KROKÓW DO SUKCESU

1. Zidentyfikuj cele publikacji i jej niezbędne elementy składowe.

Zacznij od wypisania celów swojej publikacji oraz określenia potrzeb i oczekiwań przyszłych czytelników. Jako wskazówek użyj pytań z punktów 1 i 4. Jakość i skuteczność Twoich umiejętności będzie się poprawiać proporcjonalnie do wysiłku włożonego w analizę różnych pomysłów jeszcze przed rozpoczęciem pracy.

Równoległe do wspomnianej analizy przemyśl tekst i elementy graficzne, które mają się znaleźć w Twojej publikacji. Ogłoszenie o balu sylwestrowym, które będzie wisiało na tablicy ogłoszeń na osiedlu, wymaga zupełnie innego podejścia niż wnioski do banku o sfinansowanie przedsięwzięcia gospodarczego.

Dobry projekt to informacja podana tak, że staje się oczywista. Nie pozwól, by dążenie do oryginalności wzięło górę nad podejściem praktycznym i adekwatnym do treści przekazu.

2. Przygotuj pomniejszone szkice przedstawiające planowany układ łamów oraz położenie tytułów i elementów graficznych.

Przejrzyj swoją kartotekę pomysłów – często znajdziesz tam niejedną myśl, którą warto będzie uwzględnić w przygotowywanej właśnie przez Ciebie publikacji. Nie obawiaj się plagiatu: po tym, jak dopasujesz te pomysły do aktualnych potrzeb i wprowadzasz wszystkie późniejsze poprawki, ostateczny rezultat będzie przede wszystkim Twój.

W następnej fazie wypróbuj różne możliwości rozmieszczenia tekstu i ilustracji, używając do tego ołówka i kartki z zaznaczonym układem łamów. Pracuj możliwie jak najszybciej, broniąc się przed pokusą przeredagowywania pomysłów w trakcie ich realizacji oraz zbyt dokładnego planowania. Na tym etapie Twoim celem jest wyłącznie określenie ogólnego wrażenia na temat różnych układów tekstu i ilustracji.

Przekonasz się, że do głowy przyjdzie Ci więcej pomysłów i będziesz pracował znacznie szybciej, gdy będziesz używać kartki i ołówka zamiast komputera.

3. Wybierz odpowiedni krój, stopień i styl pisma dla każdego elementu składowego architektury strony, po czym zachowaj swoje decyzje w postaci stylów.

Kiedy już ustalisz najlepszy układ strony, przenieś jego szkic do komputera. Od tej chwili Twoja publikacja zacznie nabierać życia, w miarę jak będziesz zastępować wężyki i linie ze szkicu wybranymi cechami pisma.

Jeśli jesteś zadowolony z efektów dokonanych wyborów cech pisma, zapisz je w postaci stylów. Style przechowują informacje związane z formatowaniem oraz umożliwiają łatwą zmianę wyglądu całej publikacji poprzez ich modyfikację.

Przy rozmieszczaniu elementów graficznych wykorzystaj ich zeskanowane kopie bądź zaznacz je schematycznie prostokątami.

4. Przygotuj i wydrukuj pierwszą wersję. Wykorzystaj operację „Save as” („zapisz jako”) do zapamiętania na dysku różnych alternatyw, które zbadasz.

Teraz rozpoczyna się prawdziwa praca!

Jest mało prawdopodobne, że pierwsza wersja, jaka ujrzy światło dzienne, będzie Cię w pełni zadowalać. Kiedy zobaczysz publikację w jej naturalnych rozmiarach, zwykle zauważysz wiele rzeczy, które można i należy poprawić.

Najczęstszymi poprawkami na tym etapie są:

- dopasowanie do siebie stopnia pisma i interlinii,
- eksperymenty z ograniczaniem odstępów międzyliterowych i międzywyrazowych,
- przeredagowanie tytułów i śródtytułów tak, by te same treści przekazać mniejszą liczbą słów, zajmując mniej miejsca.

Podczas przeglądania swojej publikacji, zwłaszcza wykorzystanych w niej akcentów graficznych i sposobów wyróżniania fragmentów tekstu, zadawaj

sobie stale pytanie: „czy to wyróżnienie albo akcent graficzny jest absolutnie niezbędne?”. Głównym rezultatem tej fazy przygotowania publikacji będzie najprawdopodobniej usunięcie wielu elementów, które przedtem wydawały się konieczne.

Często, nim osiągniesz zadowalający efekt końcowy, będziesz musiał przygotować wiele kolejnych wydruków.

5. Kiedy już nie masz zastrzeżeń edytorskich, spójrz na swoją publikację z punktu widzenia jej celów oraz oczekiwań czytelników określonych w kroku pierwszym.

Zadaj sobie następujące pytania:

- Czy treść jest przekazywana wystarczająco zrozumiale?
- Czy układ odzwierciedla planowaną hierarchię treści?
- W jakim stopniu ta publikacja spełnia oczekiwania odbiorców?

To, ile zyska siła wyrazu Twojej publikacji, zależy od jej analizy pod kątem funkcji, którą ma spełniać, oraz przewidywanego kręgu odbiorców – jego potrzeb i oczekiwań.

6. Przeglądaj i udoskonalaj swoją publikację tyle razy, ile będzie trzeba. Korzystaj przy tym z Podręcznej Listy Kontrolnej jako przewodnika.

Ostatni krok może zająć Ci nawet jedną czwartą czasu poświęconego na przygotowanie publikacji, ale ten nakład pracy zwraca się w postaci wyraźnie poprawionego przekazu.

Dopracowanie dokumentu w tej fazie obejmuje zagadnienia takie jak:

- eliminowanie niepożądanych odstępów wewnątrz tytułów i tekstu podstawowego – na przykład standardowych odstępów wewnątrz tytułów i zbyt dużych odstępów po znakach przestankowych,
- przegląd wyrazów przenoszonych pod kątem poprawności ich przenoszenia w zależności od kontekstu oraz wyeliminowania „luźnych” i „gęstych” wierszy,
- korekta gramatyczna dążąca do poprawienia niewłaściwych podstawień dokonanych przez program sprawdzający poprawność językową tekstu.

Powyższe sześć kroków jednoznacznie wskazuje na znaczenie planowania i powtarzania. Zawodowi redaktorzy rzadko poprzestają na pierwszym projekcie. Wręcz przeciwnie – tym, po czym można odróżnić profesjonalistę od amatora, jest gotowość poświęcenia czasu na planowanie przed rozpoczęciem pracy, otwartość na eksperymentowanie z różnymi pomysłami oraz – co najważniejsze – zdecydowane dążenie do dopracowania każdego szczegółu.

To, czy będziesz coraz lepszym redaktorem, zależy od Twojej gotowości do ciągłego planowania, eksperymentowania i poprawiania.

Podsumowując rozważania zawarte w tym rozdziale, zauważmy, że stosowane podczas redakcji tekstu operacje możemy podzielić na cztery grupy: operacje na plikach, pisanie tekstów i wprowadzanie zmian (edytowanie), formatowanie na ekranie i formatowanie wydruku. Ponadto współczesne programy redagowania tekstu zawierają rozszerzenia takie jak: sprawdzanie poprawności ortograficznej, korespondencja seryjna, spisy treści, przygotowywanie indeksów i inne.

7. ARKUSZE KALKULACYJNE

Najpierw poznajmy kilka terminów.

- [1999a] ARKUSZ KALKULACYJNY – program symulujący arkusz obliczeniowy na ekranie komputera, umożliwiający wpisywanie ukrytych wzorów, które wykonują obliczenia na danych widocznych w komórkach. Wiele arkuszy kalkulacyjnych ma rozbudowane możliwości graficzne i prezentacyjne, co pozwala tworzyć atrakcyjne materiały.
- [1999a] ARKUSZ KALKULACYJNY TRÓJWYMIAROWY – arkusz kalkulacyjny, w którym można utworzyć skoroszyt złożony z wielu stron logicznie umieszczonych jedna nad drugą, z których każda jest oddzielnym arkuszem roboczym.
- [1999a] ARKUSZ OBLICZENIOWY – w arkuszu kalkulacyjnym graficzna reprezentacja zestawienia liczbowego, utworzonego przez kolumny i wiersze, do którego można wpisywać etykiety (nagłówki i podtytuły) oraz liczby. Arkusz obliczeniowy jest tabelą złożoną z wierszy (oznaczanych liczbami) i kolumn (oznaczanych kolejnymi literami alfabetu), których przecięcia tworzą pojedyncze komórki. Każda komórka ma własny adres, np. B4 lub D19. W każdej komórce można umieścić wartość, która jest liczbą lub ukrytym wzorem wykonującym obliczenia, lub etykietę będącą nagłówkiem lub objaśnieniem. Wzór może zawierać stałe, np. 2+2, jednakże użyteczne wzory zawierają odwołania do komórek, np. D9+D10. Umieszczając wzory w komórkach arkusza obliczeniowego, można utworzyć złożoną sieć powiązań między jego częściami. Po wprowadzeniu odpowiednich wzorów można zmieniać stałe, np. wysokość stawek podatkowych lub wielkość przyspieszenia grawitacyjnego, i obserwować zmiany zależnych od nich wartości.
- [2000a] ARKUSZ KALKULACYJNY – program stosowany do obliczeń matematycznych, statystycznych i finansowych. Dane organizowane są w komórkach. Związek pomiędzy wartościami w poszczególnych komórkach definiują formuły – zmiana wartości zapisanej w jednej komórce może spowodować zmianę zawartości innej. Arkusz kalkula-

cyjny pozwala na dowolne sortowanie danych, ich graficzną prezentację za pomocą wykresów, a także drukowanie informacji. Najpopularniejszym arkuszem kalkulacyjnym jest Microsoft Excel.

7.1. TROCHĘ HISTORII

Bardzo ważne z punktu współczesnego użytkownika są programy obliczeń tablicowych, nazywane też arkuszami kalkulacyjnymi lub rachunkowymi (ang. *electronic spreadsheets* lub *financial planning programs*). Za pierwszy z programów tej klasy uznaje się Visicalc, wprowadzony w 1979 r., początkowo na komputer Apple. Jego autorami byli Dan Bricklin, Dan Fylstra, Bob Frankston. Uważa się, że dzięki niemu wzrosła sprzedaż komputerów osobistych. Następnie dużą popularność zdobył Lotus 1-2-3 firmy Lotus, a przede wszystkim jego wersja na IBM PC. Jego autorzy to Mitch Kapor i Jonathan Sachs. Został on wprowadzony w lutym 1982 r. Program ten wyznaczył standardy dla innych programów z tej grupy. Obecnie do najbardziej popularnych arkuszy kalkulacyjnych należą MS Excel, Corel QuattroPro i oczywiście Lotus 1-2-3, a kiedyś były to SuperCalc (Computer Associates) i Multiplan (Microsoft). W Polsce odpowiednikami programu Multiplan były Tabplan (CSK) i Multoplan (Mikrokomputery). Zastosowanie podobnych rozwiązań sprawia, że znajomość jednego programu umożliwia łatwe opanowanie wszystkich pozostałych.

7.2. ZASADY DZIAŁANIA

Arkusz kalkulacyjny umożliwia wprowadzanie danych liczbowych i tekstu z klawiatury na ekran reprezentujący fragment tablicy, nazywanej arkuszem roboczym (ang. *worksheet*). Użytkownik ma możliwość zdefiniowania elementu tablicy poprzez zdefiniowanie operacji arytmetycznych i zależności funkcyjnej na innych elementach tablicy. Utworzenie złożonego modelu matematycznego nie wymaga znajomości języka programowania. W tego typu programach zdefiniowanych jest wiele funkcji matematycznych, logicznych i niektóre funkcje statystyczne, które użytkownik wywołuje poprzez nazwę, wskazując podświetleniem zakres argumentów, do których się odnoszą.

Użytkownik ma możliwość prezentacji różnego typu wykresów, które wyświetlane są na ekranie i drukowane. Każda zmiana w danych liczbowych, wprowadzona z klawiatury na ekran, powoduje natychmiastowe przeliczenie wielkości zależnych i zmianę wykresu. Dzięki temu użytkownik może eksperymentować z danymi i weryfikować zaproponowany model matematyczny.

Użytkownik ma do dyspozycji *menu* przedstawione w górnej części ekranu oraz paski narzędzi, podobnie jak w redaktorze tekstu. W razie wątpliwości użytkownik może odwołać się do pomocy, a także znaleźć informacje dotyczące dostępnych funkcji matematycznych. W wypadku gdy wielkość tablicy przekracza rozmiary kartki papieru, na którym dokonywany jest wydruk, program automatycznie dzieli tabelę na części umożliwiające ich drukowanie w formacie A4.

7.3. BUDOWANIE ARKUSZY

W arkuszu kolumny oznaczane są literami od A do IV (litery I oraz V), a wiersze numerami od 1 do 8192 (co najmniej). Element arkusza leżący na przecięciu kolumny i wiersza to komórka, która może zawierać dane liczbowe, tekstowe (etykiety) albo wzory.

Do wyboru komórki użytkownik może użyć myszki lub klawiszy kierunkowych. Nawigacja w arkuszu odbywa się podobnie jak w edytorze tekstu, z tym że podstawowym elementem jest komórka, a nie znak. I tutaj przez odpowiednie kombinacje klawiszy można natychmiast wskazać lewy górny i prawy dolny róg arkusza. Także w arkuszu kalkulacyjnym można pracować w kilku oknach.

W poszczególne komórki arkusza użytkownik wpisuje liczby i teksty oraz wzory, a komputer wizualizuje je w wybranym formacie. Zawartości poszczególnych komórek możemy edytować znak po znaku albo też dowolnie zmieniać przez dokonanie nowego wpisu. We współczesnych arkuszach kalkulacyjnych mamy możliwość ochrony wybranych komórek przed zmianą zawartości.

Użytkownik może zaznaczyć dowolny prostokątny obszar w arkuszu, w szczególności jedną komórkę, jako blok, który następnie może kopiować, kasować, przenosić w inne miejsca arkusza lub zapamiętywać w pliku na dysku. Blok może być zdefiniowany przez adresy narożnych, przekątnych komórek obszaru lub przez podświetlenie.

Podczas kopiowania komórek i bloków zawierających formuły wykorzystujące zawartości innych komórek program automatycznie przedadresowuje argumenty względem nowego położenia. Dzięki temu w nowym miejscu formuła odnosi się do komórek o tym samym położeniu względem komórki, w której jest zapisana. Możemy także zastosować adresy bezwzględne, które nie ulegają zmianie przy kopiowaniu komórek.

Użytkownik definiuje formuły przez wpisanie znaku działania lub funkcji (operatora) i komórki albo ciągu komórek jako argumentów. Możemy także wskazać argumenty przez podświetlenie wybranych komórek, a to powoduje automatyczne wpisanie odpowiednich adresów do wzoru.

W arkuszu kalkulacyjnym oprócz podstawowych funkcji matematycznych, takich jak część całkowita, wartość modulo, pierwiastek kwadratowy, funkcje trygonometryczne, logarytm i jego odwrotność, generator liczb losowych, znajdują się też funkcje statystyczne, logiczne oraz tzw. operacje finansowe. Wśród funkcji statystycznych są operacje na ciągach liczb (zakresach komórek), takie jak obliczanie sumy, znajdowanie liczby argumentów, wartości minimalnej i maksymalnej, obliczanie wartości średniej, wariancji i odchylenia standardowego. Funkcje logiczne obejmują działania na wartościach logicznych typu prawda-fałsz oraz operacje warunkowe. Operacje finansowe związane są głównie z procentami składanymi. Dostępne są funkcje umożliwiające działania na danych.

Automatyczne przeliczenie wszystkich komórek powiązanych wzorami z komórką, w której zmieniono wartość, może być uciążliwe przy bardzo dużych arkuszach, dlatego też użytkownik może je wyłączyć i uruchamiać na żądanie.

7.4. ARKUSZ JAKO BAZA DANYCH

Informacje zapisane w arkuszu mogą stanowić bazę danych. W takiej interpretacji wiersze tablicy reprezentują rekordy, a komórki w kolejnych kolumnach pola bazy danych.

Arkusz kalkulacyjny zawiera podstawowe operacje sortowania całej bazy danych lub jej części. Użytkownik definiuje sortowania według rosnącego lub malejącego porządku elementów wybranej kolumny. Możemy stosować kilka kluczy sortowania, czyli sortować według kolejności elementów w jednej kolumnie, a przy takich samych wartościach elementów według kolejności elementów innej kolumny. Użytkownik ma także możliwość filtrowania danych, czyli wybrania z całego arkusza części danych spełniających określone kryteria.

Tabela 7.1 jest poglądową ilustracją tego podejścia i zawiera dane o sprzedaży czterech oddziałów pewnego przedsiębiorstwa w kolejnych kwartałach roku.

Tabela 7.1.

Oddział	1. Kw.	2. Kw.	3. Kw.	4. Kw.
Północ	90	50	65	85
Południe	50	40	45	70
Wschód	25	30	40	20
Zachód	10	20	30	45

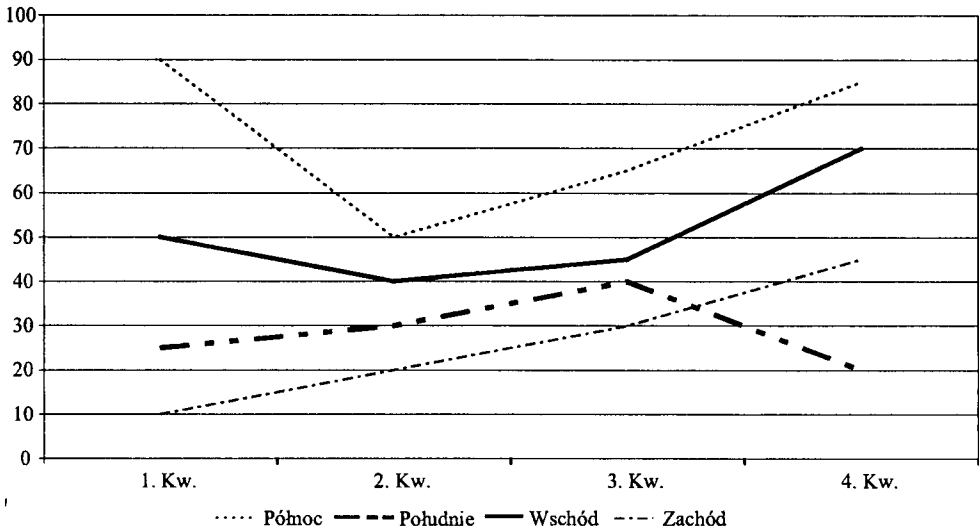
Źródło: opracowanie własne.

7.5. WYKRESY

Szeregi liczb zawarte w wierszach lub kolumnach możemy przedstawić w postaci wykresów liniowych, słupkowych, słupkowych składanych lub kołowych w wersji dwu- lub trójwymiarowej. Arkusze kalkulacyjne oprócz wymienionych najczęściej używanych typów wykresów oferują wykresy radarowe, bąbelkowe oraz kombinacje i modyfikacje wspomnianych typów wykresów. Na jednym wykresie mamy możliwość przedstawienia kilku szeregów liczbowych. Do opisu osi wykresu możemy użyć wierszy lub kolumn z tego samego arkusza. W jednym arkuszu możemy zdefiniować wiele różnych wykresów. Na rysunkach poniżej przedstawione zostały przykładowe wykresy do danych zawartych w tabeli 7.1.

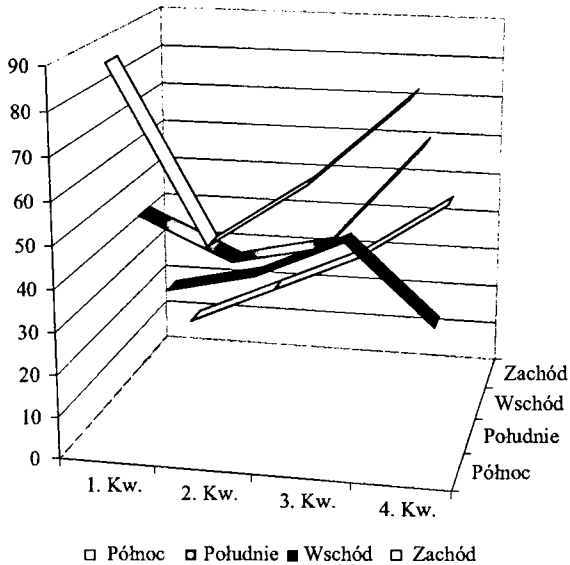
WYKRESY LINIOWE

Sprzedaż



Rysunek 7.1. Wykres liniowy płaski.

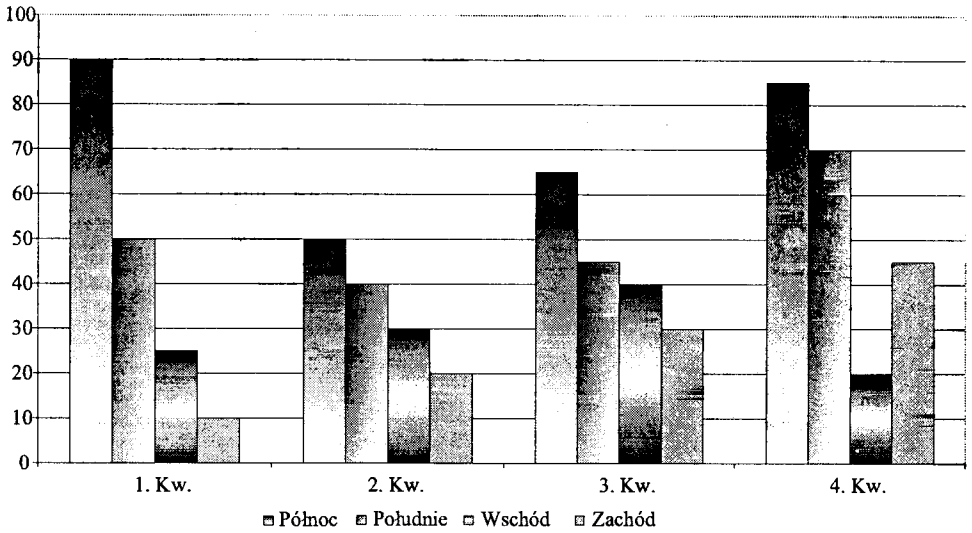
Sprzedaż



Rysunek 7.2. Wykres liniowy trójwymiarowy.

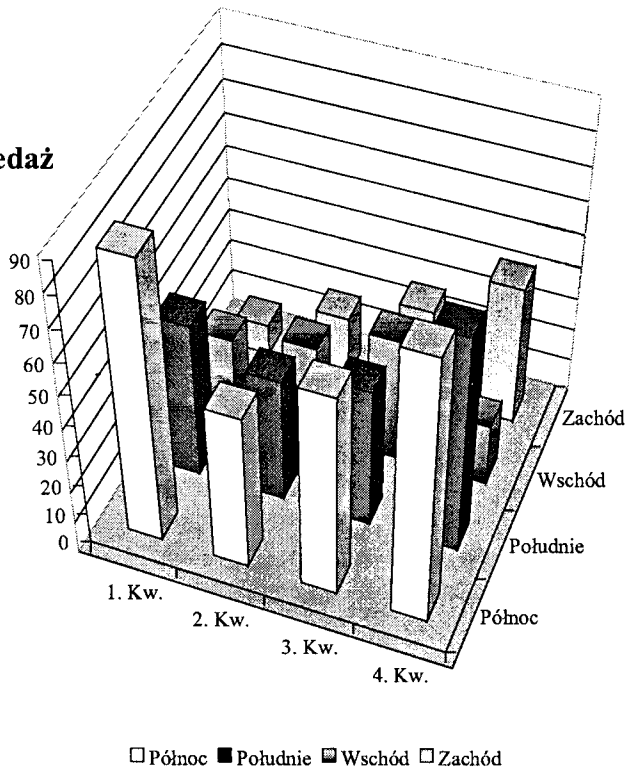
WYKRESY SŁUPKOWE

Sprzedaż



Rysunek 7.3. Wykres słupkowy płaski.

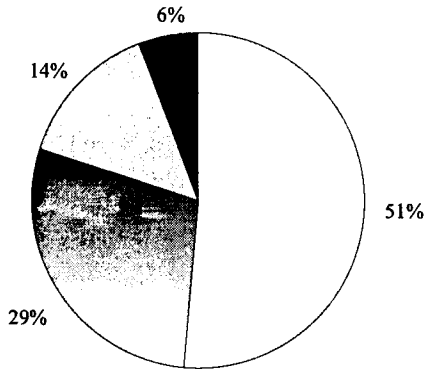
Sprzedaż



Rysunek 7.4. Wykres słupkowy trójwymiarowy.

WYKRESY KOŁOWE

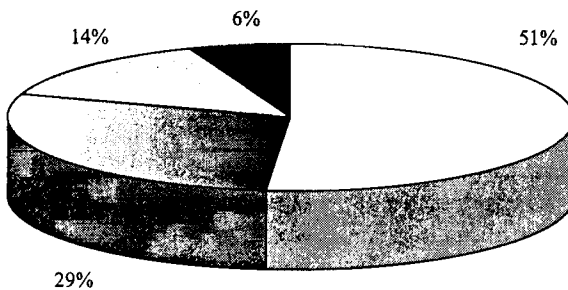
Sprzedaż



□ Północ ■ Południe ▨ Wschód ■ Zachód

Rysunek 7.5. Wykres kołowy płaski.

Sprzedaż



□ Północ ■ Południe ▨ Wschód ■ Zachód

Rysunek 7.6. Wykres kołowy trójwymiarowy.

Wykresy są skalowane automatycznie, ale istnieje możliwość skalowania ręcznego. ✓ Każdy wykres zawiera wiersze tytułu, opisy osi pionowej i poziomej, legendę do poszczególnych szeregów liczbowych, a ponadto każdy punkt wykresu może mieć własną etykietę.

7.6. MODELOWANIE

Arkusze kalkulacyjny może służyć do przedstawienia zależności pomiędzy danymi z zakresu ekonomii, techniki i innych nauk. Reprezentowane przez te zjawiska i procesy powiązania między danymi można nazwać modelem lub szablonem (ang. *template*). Przygotowany raz model (szablon) może być wykorzystany z różnymi zestawami danych, przy czym im bardziej uniwersalny jest model, tym krąg odbiorców może być szerszy. Ciągi powtarzalnych operacji, związanych operacjami na danych (komórkach arkusza), mogą być zapisane w arkuszu jako makrorozkaz (ang. *macro*) uruchamiany w razie potrzeby.

7.7. WYMIANA DANYCH MIĘDZY PROGRAMAMI

Mimo zewnętrznego podobieństwa sposób zapisu danych w arkuszach różnych producentów różni się tak istotnie, że nie można ich przenosić.

Istnieją programy konwersji, ale ich zastosowanie jest dość ograniczone. Stosunkowo łatwo jest przenieść dane liczbowe, natomiast wiele problemów sprawia przenoszenie wbudowanych w arkusz zależności pomiędzy komórkami i struktury obliczeń.

8. PROGRAMY PREZENTACYJNE

- [1999a] PROGRAM DO TWORZENIA GRAFIKI PREZENTACYJNEJ – aplikacja przeznaczona do tworzenia i doskonalenia wykresów czy zestawień, tak aby były wizualnie atrakcyjne oraz zrozumiałe dla słuchaczy. Profesjonalny pakiet do tworzenia grafiki prezentacyjnej taki jak Lotus Freelance Graphics czy Microsoft PowerPoint pozwala na tworzenie szerokiego spektrum różnorodnych diagramów i wykresów oraz umieszczanie na nich, wszędzie gdzie to jest konieczne, tytułów, legend, opisów i objaśnień. Programy do tworzenia grafiki prezentacyjnej zazwyczaj zawierają biblioteki gotowych rysunków wykorzystywanych do ożywiania wykresu przez dodawanie rysunków związanych tematycznie z przedmiotem prezentacji; np. rysunku samolotu do zestawienia ilustrującego dochody towarzystwa lotniczego. Prezentację przygotowaną za pomocą programu do tworzenia grafiki prezentacyjnej można wydrukować, naświetlić na kliszy, a także wyświetlić na ekranie w formie pokazu.

W uzupełnieniu definicji warto wspomnieć o programach Corel Presentations oraz Impress z pakietu OpenOffice, także przeznaczonych do tworzenia grafiki prezentacyjnej.

8.1. TROCHĘ HISTORII

Idea programu prezentacyjnego w jego komercyjnej wersji pochodzi od Boba Gaskina, doktoranta Uniwersytetu w Berkeley. W 1984 założył w Dolinie Krzemowej firmę pod nazwą Forethought i wspólnie z Dennisem Austinem rozwijał program pod nazwą „Presenter”. Ze względu na problemy z zastrzeżoną nazwą „Presenter” Bob Gaskin zaproponował nową – PowerPoint.

PowerPoint w wersji 1.0 był przygotowany w 1987 r. dla komputera Macintosh firmy Apple. Pracował w wersji czarno-białej, tworząc tekst i grafikę, które można było za pomocą fotokopiarki przenieść na folie. Wtedy nie było jeszcze komputerów przenośnych, takich jak notebooki, i używano rzutników do folii lub przezroczycy.

W tym samym 1987 r. Forethought i PowerPoint zostały zakupione przez Microsoft Corporation za 14 000 000 \$. W 1990 zostały wyprodukowane pierwsze wersje Windows, a PowerPoint stał się standardową częścią pakietu MS Office.

8.2. ZASADY PREZENTACJI

Przygotowując prezentację, powinniśmy zwrócić uwagę na kilka zasad, które ułatwią nam kontakt ze słuchaczami oraz przygotowanie atrakcyjnej prezentacji. Szersze omówienie psychologicznych aspektów prezentacji można znaleźć w publikacji [2004].

Jasność wypowiedzi – prezentacja przeznaczona jest dla określonych ludzi. Powinniśmy starać się z nimi rozmawiać, a nie przemawiać, używając poprawnego języka oraz unikając w miarę możliwości fachowego żargonu i obcych słów. Przydatne jest założenie, że temat jest nowy dla widzów, a prezentując go, staramy się z kolei pokazać, że jest nam dobrze znany. Mówiąc, pracujemy głosem, zmieniając jego natężenie i tempo wypowiedzi. Nieustannie obserwujemy reakcje słuchaczy i według nich modyfikujemy naszą wypowiedź.

Wiedza o widowni – przygotowując prezentację, powinniśmy wiedzieć, do kogo mówimy, czyli kim są słuchacze, czego widownia oczekuje, dlaczego słuchacze przyszli. Ta wiedza pozwoli mówić ich językiem i patrzeć ich oczami, czyli dostosować prezentację do poziomu merytorycznego i możliwości słuchaczy. Widzowie powinni czuć, że są partnerami, że prezentacja pomoże rozwiązać ich problemy, że jesteśmy im pomocni i potrzebni.

Zaangażowanie – powinniśmy pokazać, że zależy nam na tym, o czym mówimy, zachowywać się spontanicznie i przeżywać swoje wystąpienie. Aby to osiągnąć, musimy być dobrze przygotowani i bardzo dobrze znać się na przedmiocie prezentacji. Umożliwi nam to przeprowadzenie dyskusji, odpowiedzi na pytania widzów oraz pozwoli się im zaangażować.

8.3. FUNKCJE PROGRAMU PREZENTACYJNEGO

Program prezentacyjny to przede wszystkim proste narzędzie do łączenia tekstu i grafiki. Podstawowym elementem struktury pokazu jest slajd. Jego zawartość i struktura zależą od autora prezentacji. Współczesne aplikacje do tworzenia prezentacji proponują wiele szablonów kolorystyki i układu elementów, takich jak tekst i grafika.

W programie prezentacyjnym, jak w każdej aplikacji, pakietu biurowego mamy standardowe operacje na plikach, między innymi takie jak zapis i odczyt prezentacji.

Jednym z podstawowych elementów slajdu jest tekst. Program prezentacyjny wyposażony jest w podstawowe funkcje, podobnie jak w redaktorze, związane ze składem tekstu, a także korektą ortograficzną i wyszukiwaniem. Mamy możliwość stosowania różnych krojów i rozmiarów czcionek, wyrównywania tekstu na marginesach, stosowania różnych odstępów międzywierszowych, wypunktowywania i numerowania.

Kolejny element pokazu to grafika. Możemy wstawiać grafikę z pliku, ze zbioru dołączonych do programu obrazków – tzw. clipartów. Mamy możliwość używania ozdobnego tekstu, tzw. wordart, a także dodatkowych symboli graficznych w postaci dymków, strzałek, gwiazdek i linii.

Jak wynika z definicji, podstawowym zadaniem programu prezentacyjnego jest wizualizacja wykresów. W sposób naturalny wiąże się z tym możliwość przedstawiania tabel z danymi.

Współczesne programy prezentacyjne umożliwiają włączenie do prezentacji takich elementów multimedialnych, jak film i dźwięk.

Ponieważ program prezentacyjny jest nie tylko elementem pakietu biurowego, ale przede wszystkim aplikacją, wyposażony jest w język i mechanizmy tworzenia makrorozkazów

(makropoleceń). Umożliwia to użytkownikowi przygotowanie niestandardowych zachowań programu prezentacyjnego podczas pokazu.

Gotowy pokaz możemy dalej ulepszać poprzez ponowną edycję wybranych slajdów, zmianę szaty graficznej oraz zmianę kolejności ukazywania się slajdów podczas pokazu. Dużym ułatwieniem dla prowadzącego prezentację są notatki, które można powiązać z poszczególnymi slajdami lub ich elementami.

Jeżeli mówimy o pokazie, to nie możemy zapominać o dramaturgii, czyli możliwości reżyserowania pokazu. Prezentacja może być całkowicie sterowana przez prowadzącego, może przebiegać automatycznie, dyscyplinując prelegenta, lub stanowić samodzielny pokaz, dostępny np. w Internecie. Ostatni istotny element – często wykorzystywany z przesadą i niekonsekwentnie – to możliwość animacji elementów slajdu.

Programy prezentacyjne rzadko występują samodzielnie, gdyż na ogół są częścią pakietu biurowego. Stąd wynika możliwość wymiany obiektów z innymi aplikacjami pakietu, w szczególności z arkuszem kalkulacyjnym. Nie mniej ważna jest możliwość współpracy z przeglądarką internetową.

Kończąc ten krótki przegląd funkcji programu prezentacyjnego, warto przypomnieć o możliwości skorzystania z wbudowanej pomocy.

Dla ułatwienia słuchaczom śledzenia pokazu i robienia notatek prezenter ma możliwość wydrukowania wszystkich slajdów w różnych formatach. Funkcja drukowania w programie prezentacyjnym umożliwia także przygotowanie materiałów do publikacji, oferując kilka standardowych formatów, ale może też dopasować wydruk do możliwości drukarki i potrzeb użytkownika.

8.4. PRZYGOTOWANIE MATERIAŁÓW

W tej części przyjrzymy się kilku elementom wpływającym na czytelność poszczególnych slajdów. Przede wszystkim chciałbym zwrócić uwagę, że zanim przystąpimy do pracy z komputerem, powinniśmy przygotować projekt prezentacji, zgromadzić takie elementy, jak zdjęcia, rysunki, dane do wykresów, teksty, które powinny znaleźć się na poszczególnych slajdach. Dopiero potem przystępujemy do pracy z wykorzystaniem programu do tworzenia prezentacji.

Dla czytelności tekstu na ekranie duże znaczenie ma rozmiar czcionki. Nie ma jednoznacznej reguły, jaki on powinien być, ale jasne jest, że zbyt duża czcionka ogranicza ilość tekstu, który możemy umieścić, a zbyt mała utrudnia czytelność. Spotykam od czasu do czasu błędem jest umieszczenie całej strony tekstu na jednym slajdzie i odczytanie go podczas pokazu. Amerykanie zaproponowali „regułę siedmiu” określającą ilość tekstu na slajdzie. Reguła ta mówi, że powinno być nie więcej niż siedem linii tekstu po maksimum siedem słów.

Drugi ważny element czytelności tekstu to kontrast między kolorem tła a kolorem czcionki. Jeżeli mamy ciemne tło, to czcionka powinna być w kolorze jasnym. Często spotykany błąd to czerwona czcionka na granatowym tle. Tekst jest dobrze czytelny na ekranie komputera, lecz prawie wcale nieczytelny, gdy zostaje wyświetlony na ekranie przez rzutnik.

9. BAZY DANYCH

Rozdział rozpoczniemy od podania kilku definicji.

- [1999a] **BAZA DANYCH** – zbiór powiązanych informacji z pewnej dziedziny, zorganizowanych tak, aby łatwo było je przeszukiwać, wyciągać z nich wnioski czy na ich podstawie podejmować decyzje. Każdy zbiór danych spełniający te funkcje nazywamy bazą danych, nawet jeśli informacje te nie są zapisane w komputerze. Faktycznymi poprzednikami współczesnych skomplikowanych biznesowych baz danych były kartoteki przechowywane w szafach i szufladach. Informacja w bazach danych jest zazwyczaj podzielona na rekordy danych, które składają się z jednego lub więcej pól danych.
- [1999a] **REKORD DANYCH** – w programie do zarządzania bazą danych pełna jednostka powiązanych danych zapisanych w nazwanych polach danych. W wypadku bazy danych rekord danych jest czasem nazywany wierszem. Rekord danych zawiera wszystkie informacje związane z obiektem przechowywanym przez bazę danych. Większość programów wyświetla rekordy danych na dwa sposoby: jako formularze do wprowadzania danych i jako tabele danych. W programie do zarządzania relacyjną bazą danych rekordy danych są wyświetlane w formie wierszy, a każde pole danych tworzy kolumnę.
- [1999a] **POLE DANYCH** – w programie do zarządzania bazą danych obszar zarezerwowany dla określonej informacji w rekordzie danych. W programie do zarządzania relacyjną bazą danych, w której wszystkie operacje wydobywania danych zwracają tabele złożone z wierszy i kolumn, pola danych są wyświetlane jako kolumny.
- [1999a] **PROGRAM DO ZARZĄDZANIA BAZĄ DANYCH** – program użytkowy oferujący narzędzia do wydobywania, modyfikacji, usuwania i wprowadzania informacji. Taki program potrafi również założyć bazę danych, a także tworzyć wyciągi z niej i drukować je na drukarce lub wyświetlać na ekranie. W dziedzinie komputerów osobistych (PC) istnieją trzy typy programów do zarządzania bazą danych: program do zarządzania kartotekową bazą danych, do zarządzania relacyjną bazą danych oraz tekstową bazą danych.

- [1999a] SYSTEM ZARZĄDZANIA BAZĄ DANYCH (DBMS) – program, który organizuje dane w bazę danych, zapewniając przechowanie danych, ich organizację i możliwości wydobywania, czasami umożliwiając jednoczesny dostęp wielu użytkowników do wielu baz danych za pomocą pól dzielonych (zarządzanie relacyjną bazą danych).
- [1999a] STRUKTURALNY JĘZYK ZAPYTAŃ (SQL – ang. Structured Query Language) – w systemie zarządzania bazą danych (DBMS) opracowany przez firmę IBM język zapytań, który stał się *de facto* standardem zapytań zadawanych bazom danych w sieciach klient-serwer. Cztery podstawowe polecenia języka SQL (SELECT, UPDATE, DELETE i INSERT) odpowiadają odpowiednio czterem podstawowym poleceniom operowania danymi (wydobywanie danych, modyfikowanie danych, usuwanie danych i wstawianie danych). Zapytania SQL przypominają strukturę pytań języka naturalnego (angielskiego). Wynik zapytania jest wyświetlany jako tabela danych składająca się z kolumn (odpowiadających polom danych) i wierszy (odpowiadających rekordom danych).
- [1999a] OBIEKTOWY INTERFEJS BAZ DANYCH (ODBC) – standard opisujący dostęp programów (w tym przeglądarki WWW) do różnych programów obsługi baz danych za pomocą zstandaryzowanego zestawu zapytań w języku SQL.

Ogólnie znane systemy baz danych wywodzą swe nazwy od ich producentów np. ORACLE – Oracle Corporation, SYBASE – Sybase INC. Oracle był pierwszą dużą firmą, która zastosowała SQL jako standardowy język zapytań. Inne znane systemy zarządzania relacyjnymi bazami to Informix i Progress. Nie zawsze nazwa pochodzi od producenta, np. DB2 to system zarządzania bazą danych firmy IBM, dBASE było produkowane przez firmę Ashton-Tate. Oddzielną grupą są specjalizowane numeryczne bazy danych. Najbardziej znanym przykładem są cyfrowe systemy informacji przestrzennej – GIS.

Programami obsługi baz danych (ang. *file management systems, data management systems, information management systems*) będziemy nazywali takie programy, które umożliwiają przechowywanie, uaktualnianie i wyszukiwanie informacji oraz drukowanie raportów. Programy zarządzania bazami danych (ang. DBMS – *database management systems*) są podstawą najważniejszych zastosowań komputerów w przedsiębiorstwach, między innymi w takich dziedzinach jak zarządzanie finansami i kadrami (w tym systemem finansowo-księgowym i kadrowo-płacowym), obsługa biblioteczna, dokumentacyjna. W dalszej części będzie używany polski termin – system zarządzania bazą danych, w skrócie SZBD.

9.1. RYS HISTORYCZNY OPROGRAMOWANIA BAZ DANYCH

Na początku lat sześćdziesiątych powstały pierwsze systemy zarządzania danymi mające służyć rozwiązywaniu konkretnych problemów firmy. W połowie tej dekady stał się dostępny jeden z pierwszych pakietów baz danych przeznaczony do użytku ogólnego przygotowany przez General Electric Company (GEC), o nazwie Integrated Data Store (IDS). Działał wyłącznie na maszynach GEC, a jego głównym autorem był Charles Bachman. Ze względu na pomysł przeniesienia IDS na nowy system komputerowy IBM360. Godrich i John Cullinane utworzyli firmę o nazwie Cullinane, później Cullinet, w której stworzono SZBD IDMS.

W 1969 r. pod patronatem CODASYL (Conference on Data Systems Languages) została opracowana specyfikacja wspólnych narzędzi baz danych, wzorowana na IDS i IDMS.

W roku 1970 badacze z IBM, bazując na pracach innego naukowca z tej firmy – E.F. Codda, zbudowali prototyp relacyjnego SZBD o nazwie System/R.

Prowadzone na wielu uniwersytetach w latach 70. i na początku 80. badania znacząco przyczyniły się do rozwoju relacyjnych baz danych. W 1983 r. IBM zaprezentował DB2, swoją pierwszą relacyjną bazę danych dla dużych komputerów.

Niewątpliwie znaczenie dla popularyzacji systemów relacyjnych baz danych miały systemy tworzone dla komputerów klasy PC, np. dBASE II firmy Ashton-Tate.

Program dBASE II, którego autorem jest C. Wayne Ratliff, powstał pod koniec lat 70. w ramach programu kosmicznego Mariner, przeniesiony został potem na mikrokomputery ośmiobitowe pracujące pod systemem operacyjnym CP/M, a następnie, w 1981 r., na komputery szesnastobitowe typu IBM PC. W 1984 r. wyprodukowano udoskonaloną wersję: dBASE III, a wkrótce dBASE III Plus i dBASE IV. Dane, język zapytań oraz format plików systemu dBASE zostały zaimplementowane w innych komercyjnych produktach, takich jak FoxPro firmy Microsoft i CA-dbFAST firmy Computer Associates. Program dBASE jest interpreterem, w związku z tym wykonanie programów użytkowych zawierających dużą liczbę rozkazów lub rozkazy powtarzane wielokrotnie w pętli trwa dość długo i aby je przyspieszyć, warto przetłumaczyć rozkazy programu na kod maszynowy. Do tego celu służą kompilatory, np. program Clipper firmy Nantuckett. Skompilowane programy użytkowe działają od 2 do 20 razy szybciej niż programy interpretowane. Po skompilowaniu tworzony jest program, którego uruchomienie nie wymaga posiadania przez użytkownika programów dBASE lub Clipper. Program działa szybciej, ale użytkownik nie może go samodzielnie modyfikować.

W ciągu ostatnich lat na kształt systemów baz danych wpływają trzy trendy: przetwarzanie wbudowane w bazę danych, obiektowość i równoległość.

Początkowo baza danych przechowywała po prostu dane, których przetwarzanie wykonywały programy użytkowe pracujące poza bazą danych. Cechą nowoczesnych relacyjnych SZBD jest przetwarzanie w samej bazie danych. Jest to istotne przy stosowaniu architektury klient-serwer.

W ostatnich latach pojawiło się wiele komercyjnych obiektowych SZBD. Producenci relacyjnych systemów wyposażają swoje produkty w obiektowe cechy, aby umożliwić korzystanie ze złożonych danych.

Pomysł stosowania równoległego przetwarzania w bazach danych nie jest nowy, ale dopiero obecnie zaczął być poważnie brany pod uwagę w systemach komercyjnych.

9.2. WŁAŚCIWOŚCI BAZY DANYCH

Rozważmy pokrótce, jakie cechy powinna mieć baza danych. Wymieńmy je: współdzielenie danych, integracja danych, integralność danych, bezpieczeństwo danych, abstrakcja danych i niezależność danych.

Współdzielenie danych – dane przechowywane w bazie danych są używane przez więcej niż jedną osobę, czasem w tej samej chwili. SZBD musi zapewnić dostęp do nich w każdej chwili.

Integracja danych – współdzielenie danych jest bardzo korzystne, ale tylko wtedy, gdy przy użytkowaniu bazy danych jest zapewniona integracja danych. Celem integracji jest unikanie niepotrzebnie powtarzających się lub zbędnych danych. Realizuje się to między innymi przez przechowywanie jednego logicznego elementu danych tylko w jednym miejscu.

Integralność danych – integralność jest drugą konsekwencją współdzielenia danych. Na bazę danych nakłada się wymaganie wiernego odwzorowania obszaru analizy rzeczywistości, dla której konkretna baza danych jest modelem. Integralność oznacza, że jeżeli w rzeczywistym świecie istnieją związki między obiektami reprezentowanymi przez dane w bazie danych, to zmiany zachodzące w rzeczywistości w jednej części takiego związku, które wpływają na inne części tegoż związku, powinny być dokładnie odzwierciedlone w zmianach zachodzących w bazie danych w części modelującej określony związek.

Bezpieczeństwo danych – zabezpieczenie bazy danych jest podstawowym sposobem zapewnienia jej integralności. Podstawową stosowaną współcześnie metodą zabezpieczenia bazy danych jest ograniczenie dostępu do niej, określenie, z pewną szczegółowością, w odniesieniu do całej lub pewnej części bazy danych, zbiorów upoważnionych użytkowników.

Abstrakcja danych – tworząc bazę danych, staramy się odwzorować w niej pewien fragment otaczającej nas rzeczywistości. Baza danych może być traktowana jako model rzeczywistości. Informacje zapisywane w bazie danych reprezentują wybrane własności interesujących nas obiektów w świecie rzeczywistym. Mówimy tu o wybranych, gdyż w żadnej bazie danych nie możemy przechowywać wszystkich właściwości obiektów ze świata rzeczywistego. W tym sensie baza danych jest modelem rzeczywistości, abstrakcją prawdziwego świata.

Niezależność danych – konsekwencją abstrakcji jest idea oddzielenia danych od procesów, które używają tych danych. Dążymy tutaj do sytuacji, w której struktura danych jest niezależna od korzystających z nich aplikacji. Rzecz w tym, aby zamiana części bazy nie wymagała zmiany korzystających z tych danych aplikacji. Także odwrotnie – zamiana części aplikacji nie wymagała modyfikacji struktury danych używanych przez tę aplikację.

9.3. SYSTEM ZARZĄDZANIA BAZĄ DANYCH

System zarządzania bazą danych (SZBD) jest powłoką, która otacza bazę danych. SZBD zawiera narzędzia umożliwiające dostęp i zarządzanie bazą danych lub ich większą liczbą. Funkcje realizowane przez większość systemów zarządzania bazą danych możemy podzielić na trzy grupy:

1. Operacje na plikach:
 - a) dodawanie nowych plików,
 - b) usuwanie plików,
 - c) zmiana struktury istniejących plików,
 - d) dopisywanie nowych danych do istniejących plików,
 - e) poprawianie danych w istniejących plikach,
 - f) usuwanie danych z istniejących plików.
2. Operacje na informacji:
 - a) pobranie na potrzeby użytkowników danych z istniejących w bazie plików,
 - b) pobieranie danych na potrzeby programów użytkowych.
3. Zarządzanie bazą danych:
 - a) rejestrowanie i śledzenie użytkowników bazy danych,
 - b) ograniczanie dostępu do informacji w bazie danych,
 - c) śledzenie pracy samej bazy danych.

We współczesnych SZBD rozróżniamy jądro systemu i zestaw narzędzi. Jądro SZBD jest podstawową częścią systemu, która realizuje wymienione powyżej grupy funkcji. Ale jest ono również odpowiedzialne za współdzielenie i integrację danych. Zestaw narzędzi SZBD zawiera zarówno oprogramowanie wbudowane jako część SZBD, jak i dostarczane przez innych producentów; są to na przykład arkusze kalkulacyjne, języki czwartej generacji, programy strojenia i monitorowania działania. Na ogół pomiędzy jądrem a zestawem narzędzi jest określony specjalny interfejs.

9.4. KARTOTEKOWE BAZY DANYCH

Kartotekowa baza danych (ang. *flat-file database*, *card-file*) jest to przechowywany na dysku plik, składający się z odrębnych elementów o identycznej strukturze, lecz różnej zawartości. Każdy z tych elementów stanowi rekord bazy danych. Rekordy dzieli się na pola, które mogą zawierać tekst, liczby, daty lub wzory. Każde pole ma swoją nazwę. Rekordy w bazie składają się z pól o tych samych nazwach i tego samego typu, ale różnią się zawartością pól.

Jako przykład takiej bazy danych może służyć zbiór kart katalogowych książek. Jedna karta stanowi rekord, a pozycje karty, takie jak np. autor i tytuł, są polami.

Program obsługujący kartotekową bazę danych umożliwia wpisywanie, edycję i usuwanie rekordów wraz z ich zawartością, a także sortowanie rekordów według wybranych pól.

Raport z takiej bazy danych definiuje się poprzez formularze (szablony) wydruku. Użytkownik może korzystać ze zdefiniowanych przez producenta typowych formularzy lub definiować własne. Raport można obejrzeć na ekranie lub wydrukować.

9.5. RELACYJNE BAZY DANYCH

Bazę nazwiemy relacyjną w sytuacji, gdy mamy przynajmniej dwa zbiory informacji połączonych wspólną daną.

Przedstawia się je na ogół w postaci tablic (nazywanych relacjami), powiązanych między sobą przez kolumny zawierające wspólne dane. Wybranie rekordu (wiersza) w jednej z tablic umożliwia wybranie poprzez kolumny zawierające wspólne dane właściwego rekordu w innej tablicy. Jedną dużą bazę danych można podzielić na mniejsze, a następnie łączyć informacje w miarę potrzeby. Bazy relacyjne są z natury strukturami wieloplikowymi.

9.6. INNE TYPY BAZ DANYCH

Wspomnimy tylko pobieżnie o bazach hierarchicznych, sieciowych, bazach o formacie swobodnym. W bazach hierarchicznych występuje pionowe powiązanie między plikami, podobnie jak to ma miejsce w strukturze katalogów i plików systemu operacyjnego MS Windows. W sieciowych bazach danych, tak jak w hierarchicznych, występują powiązania pionowe, a oprócz tego poziome lub skośne.

Bazy danych o formacie swobodnym zbudowane są z dowolnych informacji tekstowych i liczbowych bez podziału na pola. Informacja opisana jest za pomocą słów kluczowych (indeksów), które umożliwiają jej odnalezienie. W ostatnich latach bazy te zyskały na znaczeniu, szczególnie w Internecie.

Współczesne bazy danych zapewniają wielodostęp, czyli równoczesną pracę wielu użytkowników w sieci. Najbardziej znane przykłady to: MS Access firmy Microsoft, Informix (Relational Database Systems), Sybase, Oracle. Wszystkie one wykorzystują język SQL (Standard Query Language).

9.7. MODELE DANYCH

Przez model danych możemy rozumieć ogół zasad posługiwania się danymi. Mówiliśmy wcześniej o relacyjnych, hierarchicznych, obiektowych bazach danych, rozumiejąc przez to, że ich podstawą jest odpowiedni model danych.

Określając model danych, możemy wyróżnić trzy podstawowe części:

1. Definicja danych, czyli reguły określające strukturę danych.
2. Operacje na danych, czyli reguły określające posługiwanie się danymi.
3. Integralność danych, czyli reguły określające poprawne stany bazy danych.

Baza danych i związany z nią SZBD zbudowane są według zasad pewnego modelu danych. Zatem w relacyjnej bazie danych, opartej na relacyjnym modelu danych, definicja danych związana jest z pojęciem relacji, operacje na danych z zestawem operatorów relacyjnych, a integralność danych opiera się na integralności encji i integralności referencyjnej.

Przez **encję** rozumiemy tu obiekt w bazie danych, który odpowiada istniejącej w rzeczywistości i jednoznacznie identyfikowanej rzeczy.

Istnieją trzy generacje modeli danych (według literatury przedmiotu [1998a]):

1. Proste modele danych. W tym podejściu obiekty są reprezentowane za pomocą struktury rekordów zgrupowanych w strukturach plików.

- Głównymi dostępnymi operacjami są operacje na rekordach – odczytaj rekord i zapisz rekord.
2. Klasyczne modele danych. Są to hierarchiczne, sieciowe i relacyjne modele danych. Hierarchiczny model danych jest rozszerzeniem opisanego powyżej prostego modelu danych. Sieć jest rozszerzeniem podejścia hierarchicznego. Relacyjny model danych jest zasadniczym odejściem od modeli hierarchicznego i sieciowego.
 3. Semantyczne modele danych. Głównym problemem związanym z klasycznymi modelami danych, takimi jak relacyjny model danych, jest to, że zachowują one podstawową orientację opartą na rekordach. Mówiąc inaczej, znaczenie informacji w Bazie Danych – ich semantyka – nie daje się łatwo odczytać z samej bazy danych, przy użyciu podejścia klasycznego. Z tego powodu zaproponowano wiele tak zwanych semantycznych modeli danych. Semantyczne modele danych (SMD) próbują dostarczyć bardziej znaczących sposobów reprezentowania znaczenia informacji, niż to jest możliwe przy modelach klasycznych. Pod wieloma względami obiektowy model danych może być uważany za semantyczny model danych.

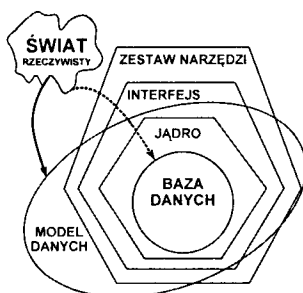
9.8. BUDOWA BAZY DANYCH

Jak wspomniano wcześniej, baza danych jest modelem fragmentu rzeczywistości. Zatem ważnym elementem budowania bazy danych jest proces modelowania.

Na rysunku 9.1. przedstawiono w sposób schematyczny SZBD i wzajemne nakładanie się związanych z nim elementów występujące w tym rozdziale.

W procesie tworzenia bazy danych jako modelu musimy uzyskać od użytkowników początkowy zbiór informacji i wymagań dotyczących przetwarzania danych. Następnie określa się zawartość bazy danych, niezależnie od wymogów konkretnej implementacji fizycznej.

Z budową bazy danych wiąże się termin analiza danych. Spotykamy dwa uzupełniające się sposoby wykonania analizy danych – normalizację i modelowanie związków encji. Tu, podobnie jak wcześniej, encję rozumiemy jako reprezentację w modelu jakiegoś rzeczywistego jednoznacznie identyfikowanego obiektu. Normalizacja jest czasami charakteryzowana jako metoda projektowania „z dołu do góry”, a modelowanie związków encji jako metoda projektowania „z góry na dół”.



Rysunek 9.1. SZBD a baza danych i model danych.

10. SZTUCZNA INTELIGENCJA

- [1999a] SZTUCZNA INTELIGENCJA (AI) – dziedzina informatyki, która zajmuje się usprawnianiem komputerów przez zaprogramowanie im cech charakterystycznych dla ludzkiej inteligencji, jak np. rozumienie języka naturalnego oraz podejmowanie decyzji w warunkach niepewności.
- [1999a] SYSTEM EKSPERTOWY – program przetwarzający wiedzę z pewnej dziedziny i służący do rozwiązywania problemów i podejmowania decyzji. System ekspertowy zawiera bazę wiedzy, zwykle wyrażoną w formie reguł JEŻELI/TO oraz narzędzia do udzielania odpowiedzi na podstawie informacji z tej bazy. System zadaje serię pytań, aby zorientować się w sytuacji, po czym stawia swoją diagnozę. Większość systemów ekspertowych uzupełnia swoją odpowiedź współczynnikiem pewności, wahającym się od „przypuszczenia” do „naukowej pewności”.
- [1999a] PROLOG – język programowania wysokiego poziomu, używany w badaniach nad sztuczną inteligencją i w stosujących ją aplikacjach, zwłaszcza w systemach ekspertowych. PROLOG, którego nazwa jest skrótem od *PROgramming in LOGic*, jest językiem deklaratywnym, tzn. koncentrującym się na opisie rozwiązywanego problemu, a nie na tworzeniu procedur służących do jego rozwiązania. Przypomina języki zapytań stosowane w systemach zarządzania bazą danych, np. strukturalny język zapytań (SQL – ang. Structured Query Language). PROLOG-u można używać do zadawania pytań w rodzaju „Czy Foster City znajduje się w Kalifornii?”. Istnieje jednak zasadnicza różnica między PROLOG-iem a systemami zarządzania bazą danych. Bazy danych przechowują dane, które można wydobywać, programy napisane w PROLOG-u natomiast przechowują wiedzę wykorzystywaną do wnioskowania o tym, co jest prawdziwe, a co nie.
- [1997] INTELIGENCJA SZTUCZNA (ang. Artificial Intelligence – AI) – dział informatyki, którego przedmiotem jest badanie reguł rządzących tzw. inteligentnymi zachowaniami człowieka, tworzenie modeli formalnych tych zachowań i – w rezultacie – programów komputerowych symulujących te zachowania. Do zachowań inteligentnych

należy postrzeganie, rozpoznawanie, uczenie się, operowanie symbolami, posługiwanie się językiem, rozwiązywanie problemów, twórczość i inne. Programy komputerowe stosuje się w celach eksperymentalnych, a także praktycznych, takich jak: rozpoznawanie kształtów (np. liter, rysunków, zdjęć), dźwięków (np. mowy), prowadzenie gier (np. szachy), dowodzenie twierdzeń, komponowanie muzyki, tłumaczenie z jednego języka naturalnego na inny, formułowanie ekspertyz, diagnoz lekarskich, sterowanie robotami i inne. Sztuczna inteligencja wytworzyła wiele metod i narzędzi – zarówno konceptualnych (np. języki programowania, głównie Lisp i Prolog, algorytmy wyszukiwania informacji, dopasowywania wzorców i inne), jak i materialnych (komputery o specjalnej architekturze, urządzenia komunikowania się z komputerem i inne). Metody sztucznej inteligencji wpłynęły na rozwój urządzeń zw. inteligentnymi robotami, stosowanych w niektórych dziedzinach nauki i gospodarki, głównie w przemyśle. Sztuczna inteligencja ma liczne powiązania z psychologią, fizjologią, bioniką, cybernetyką, teorią gier i innymi dziedzinami, których pojęcia, metody i wyniki wykorzystuje w konstruowaniu programów i urządzeń symulujących zachowania inteligentne. Wzbogaca też te dziedziny poprzez oferowanie im własnych pojęć i aparatu badawczego.

Sztuczna inteligencja jest rozwijającą się od wielu lat dyscypliną badawczą z zakresu nauk komputerowych. W obu przytoczonych definicjach sztucznej inteligencji zauważa się, że jest ona dziedziną informatyki zajmującą się komputerową symulacją inteligentnego zachowania.

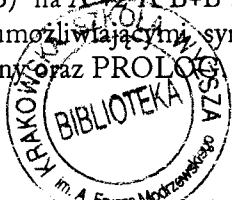
Warto tu wspomnieć ponownie o wymienionej w rozdziale 2 Augustie Adzie lady Lovelace, która pierwsza pisała o możliwości użycia maszyny analitycznej do komponowania muzyki oraz o arytmetyce systemu nerwowego, co można uznać za pierwsze idee sztucznej inteligencji.

Przykładami zagadnień znajdujących się w kręgu zainteresowań Sztucznej Inteligencji są między innymi: przetwarzanie języków naturalnych, rozpoznawanie obrazów, uczenie się, automatyczne dowodzenie twierdzeń, systemy ekspertowe, sieci neuronowe.

10.1. PROGRAMY OPERUJĄCE NA SYMBOLACH

Tradycyjne języki programowania są dostosowane do operacji na liczbach i próba obliczenia wyrażenia, w którym jednej ze zmiennych nie przypisano uprzednio wartości liczbowej, sygnalizowana jest jako błąd. Korzystając np. z BASIC-u, bez trudu otrzymamy wynik działania postaci $(1+2)^2 = 9$, podczas gdy próba przekształcenia wyrażenia $(A+B)^2$ będzie nieudana. Tymczasem człowiek bez trudu potrafi przekształcić wyrażenie symboliczne $(A+B)^2$ na $A^2 + 2*A*B + B^2$.

Językami umożliwiającymi symboliczne przekształcenie wyrażenia są LISP i różne jego odmiany oraz PROLOG. Mamy też takie systemy jak MATHEMATICA i MathCAD.



Zakres operacji symbolicznych wykonywanych przez systemy operacji na symbolach jest bardzo rozbudowany. W zastosowaniach praktycznych ograniczeniem jest bardziej czas oczekiwania na wynik niż brak jakiegoś rodzaju operacji. Podstawowe możliwości oferowane przez te systemy to:

- operacje na wyrażeniach arytmetycznych w zakresie dodawania, odejmowania, mnożenia, dzielenia i potęgowania liczb wymiernych,
- operacje algebraiczne,
- rozwiązywanie równań,
- operacje macierzowe,
- rozwiązywanie układów równań,
- upraszczanie wyrażeń trygonometrycznych i hiperbolicznych,
- różniczkowanie i szeregi Taylora,
- całkowanie,
- granica funkcji,
- rozwiązywanie równań różniczkowych zwyczajnych pierwszego i wyższych rzędów.

10.2. PROGRAMY WSPOMAGANIA DECYZJI

Programy wspomaganie decyzji, zwane też systemami ekspertowymi, mają za zadanie podczas interaktywnej „rozmowy” z użytkownikiem postawić diagnozę w sprawie, w której program jest ekspertem.

System ekspertowy zbudowany jest z tzw. bazy wiedzy, mającej postać zbioru twierdzeń typu „jeśli..., to...”, modułu wnioskującego przetwarzającego nabywane informacje, modułu wyjaśniającego decyzje oraz modułu komunikacji z użytkownikiem.

Do dziś na świecie zrealizowano wiele systemów ekspertowych. Można je podzielić na dwie grupy – problemowo-orientowane i ogólne (tzw. *expert system shells*). W drugim przypadku musimy dołączyć bazę wiedzy dotyczącą w zasadzie dowolnej dziedziny wiedzy.

10.3. ROZMOWA Z KOMPUTEREM

Badania nad zastosowaniem języka naturalnego w konwersacji z maszyną są prowadzone od wielu lat. Klasycznym już przykładem opisanym w książce [1976b] jest system ELIZA (nazwa nawiązuje do imienia bohaterki *Pigmaliona* Bernarda Shawa). System ten zaprogramowano w MIT (Massachusetts Institute of Technology) przy realizacji projektu związanego z komunikacją w języku naturalnym między człowiekiem i maszyną. W przytoczonym poniżej dialogu urządzenie końcowe odgrywa rolę psychiatry prowadzącego wywiad z pacjentem:

DZIEWCZYNA:

WSZYSCY MEŹCZYŹNI SĄ JEDNAKOWI.

URZĄDZ. KOŃCOWE:

JAK TO JEST MOŻLIWE?

DZIEWCZYNA:

ONI ZAWSZE CZEGOŚ OD NAS CHCĄ.

- URZĄDZ. KOŃCOWE: CZY MOŻESZ SOBIE PRZYPOMNIEĆ JAKIŚ KONKRETNY PRZYKŁAD?
- DZIEWCZYNA: OTÓŻ MÓJ CHŁOPIEC SPROWADZIŁ MNIE TUTAJ.
- URZĄDZ. KOŃCOWE: TWÓJ CHŁOPIEC SPROWADZIŁ CIEBIE TUTAJ?
- DZIEWCZYNA: POWIEDZIAŁ, ŻE ZBYT CZĘSTO JESTEM PRZYGNĘBIONA.
- URZĄDZ. KOŃCOWE: PRZYKRO MI SŁYSZEĆ, ŻE JESTEŚ PRZYGNĘBIONA.
- DZIEWCZYNA: TO JEST PRAWDA, JESTEM NIESZCZĘŚLIWA.
- URZĄDZ. KOŃCOWE: CZY MYŚLISZ, ŻE TWOJE PRZYJŚCIE TUTAJ UCZYNI CIĘ BARDZIEJ SZCZĘŚLIWĄ?
- DZIEWCZYNA: POTRZEBUJĘ POMOCY, TO JEST CHYBA OCZYWIŚTE.
- URZĄDZ. KOŃCOWE: DO CZEGO TA POMOC JEST CI POTRZEBNA?
- DZIEWCZYNA: BYĆ MOŻE POTRAFIĘ NAUCZYĆ SIĘ WYTRZYMYWANIA Z MOJĄ MATKĄ.
- URZĄDZ. KOŃCOWE: OPOWIEDZ MI WIĘCEJ O SWOJEJ RODZINIE.
- DZIEWCZYNA: OPIEKUJE SIĘ MNĄ MOJA MATKA.
- URZĄDZ. KOŃCOWE: KTO JESZCZE Z TWOJEJ RODZINY OPIEKUJE SIĘ TOBĄ?
- DZIEWCZYNA: MÓJ OJCIEC.
- URZĄDZ. KOŃCOWE: TWÓJ OJCIEC?
- DZIEWCZYNA: JESTEŚ W PEWIEN SPOSÓB PODOBNY DO MOJEGO OJCA.
- URZĄDZ. KOŃCOWE: JAKIE TY WIDZISZ PODOBIENSTWO?
- DZIEWCZYNA: NIE JESTEŚ ZBYT AGRESYWNY, ALE MYŚLĘ, ŻE NIE CHCESZ, ŻEBYM TO ZAUWAŻYŁA.
- URZĄDZ. KOŃCOWE: CO MIAŁAŚ NA MYŚLI, MÓWIĄC, ŻE NIE JESTEM ZBYT AGRESYWNY?
- DZIEWCZYNA: NIE DYSKUTUJESZ ZE MNĄ.
- URZĄDZ. KOŃCOWE: DLACZEGO UWAŻASZ, ŻE NIE DYSKUTUJĘ Z TOBĄ?
- DZIEWCZYNA: BOISZ SIĘ MNIE.
- URZĄDZ. KOŃCOWE: CZY SPRAWIA CI TO PRZYJEMNOŚĆ, ŻE JA SIĘ CIEBIE BOJĘ?
- DZIEWCZYNA: MÓJ OJCIEC BOI SIĘ WSZYSTKICH.
- URZĄDZ. KOŃCOWE: CO JESZCZE CI PRZYCHODZI DO GŁOWY, GDY MYŚLISZ O SWOIM OJCU?
- DZIEWCZYNA: TYRANIA.
- URZĄDZ. KOŃCOWE: CZY TO MA JAKIKOLWIEK ZWIĄZEK Z TYM, ŻE TWÓJ CHŁOPIEC SPROWADZIŁ CIĘ TUTAJ?

Ten przykład znacznie lepiej ilustruje niebezpieczeństwa, które wiążą się z rozmową w języku naturalnym, niż potencjalne możliwości. Występująca w przykładzie dziewczyna, użytkownik urządzenia końcowego, myliła się, sądząc, że maszyna rozumie jej wypowiedzi. W programie wszystko sprowadzało się do rozpoznania słów kluczowych i dokonania ich odpowiedniej transformacji przed wyprowadzeniem odpowiedzi. Wszystko przypomina iluzjonistyczny trik.

W systemie konwersacyjnym ELIZA program rozpoznaje ograniczony zasób słów i w przypadku słowa z tego zbioru wykonuje określone czynności. Program korzysta z tezaursusa – słownika wyrazów bliskoznacznych, który umożliwia skojarzenie różnych słów i zwrotów o tym samym znaczeniu. Taki mechanizm wydaje się wystarczający do zastosowania w systemach handlowych.

Niebezpieczeństwo takiego dialogu tkwi w tym, że rozmówca może odnieść mylne wrażenie, że maszyna rozumie więcej niż w rzeczywistości. Zaletą tego systemu był brak ograniczenia użytkownika do podzbioru języka. System raczej nakłaniał go do zachowania swobody wypowiedzi w języku angielskim, z której korzystał w codziennej konwersacji. ELIZĘ zaprogramowano, aby ukryć przed użytkownikiem te przypadki, w których system go nie rozumie.

Zauważmy, że zamiar uzyskania maksymalnej płynności dialogu stoi w sprzeczności z dążeniem do uzyskania maksymalnie precyzyjnych wypowiedzi. Dowolność spotykana w normalnej konwersacji może wykraczać poza możliwości maszyny. Przyczyną jest nieprecyzyjność oraz wieloznaczność mowy ludzkiej.

Od zbudowania ELIZY minęło ponad 25 lat, a nadal nie ma komercyjnych systemów swobodnie rozmawiających z człowiekiem na zadany temat, np. o pogodzie.

10.4. AUTOMATYCZNE TŁUMACZENIE

Badania nad rozumieniem języka naturalnego przez komputer, w szczególności prowadzenie rozmowy na dowolny temat, były podstawą do zbudowania automatycznych systemów tłumaczących. Mimo upływu lat i wzrostu mocy obliczeniowej komputerów automatyczne tłumaczenia pozostawiają wiele do życzenia. Oto kilka przykładów przygotowanych z wykorzystaniem wielojęzycznego tłumacza firmy InterTran™ dostępnego na stronie internetowej <http://wlanet.pl/poznaj/translator/>. Wszystkie przykłady zostały przygotowane tą samą metodą. Zdanie w języku polskim było tłumaczone na język obcy, a uzyskany wynik tłumaczony ponownie na język polski. Użyłem dwóch języków: angielskiego i niemieckiego.

Przykład 1.

Zielona kaczka pływa po stawie.

Język angielski: *Herbalist duck swims when pond.*

Język polski ponownie: *Zielarz kaczka pływa podczas staw.*

Język niemiecki *Grünlich Ente schwimmt zu gibt aus*

Język polski ponownie: *Zielonawy kaczka dziennikarska pływa do wydaje.*

Przykład 2.

Chytry dwa razy traci.

Język angielski: *Tricky twice loses.*

Język polski ponownie: *Podstępny dwa razy traci.*

Język niemiecki: *Hinterlistig zweimal verwirkt.*

Język polski ponownie: *Podstępny dwa razy traci.*

Przykład 3.

Wiem, że nic nie wiem. (Sokrates)

Język angielski: *I see, that small potatoes you have got me there*

Język polski ponownie: *Rozumiem, ów płotki złapałeś mnie.*

Język niemiecki: *Wiem, daß null ungebunden wiem.*

Język polski ponownie: *Wiem, że nic nigdy wiem.*

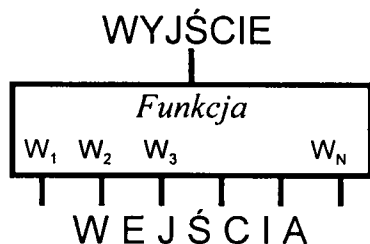
10.5. SIECI NEURONOWE

Sieci neuronowe stanowią bardzo uproszczony, a przez to łatwiejszy do skonstruowania na komputerze model rzeczywistego biologicznego systemu nerwowego, a w szczególności pewnych fragmentów naszego mózgu.

Sieci neuronowe bywają skuteczne jako narzędzia w rozwiązywaniu takich zadań, z którymi typowe komputery i typowe programy nie bardzo sobie radzą. Powody w zasadzie są dwa – obliczenia są wykonywane równolegle i sieci nie trzeba programować. Równoległość wynika z faktu, że tysiące sztucznych neuronów składających się na sieć wykonuje obliczenia równocześnie. Sieć neuronowa uczy się, co pozwala uzyskać skuteczne działanie nawet w sytuacji, gdy nie znamy algorytmu, rozwiązania zadania.

Podstawowym budulcem sieci neuronowych są sztuczne neurony. Pierwowzorem dla nich są komórki nerwowe, z których zbudowany jest mózg człowieka. Elementy używane w technicznych sieciach neuronowych zawierają tylko podstawowe właściwości swoich pierwowzorów. Ponieważ są bardzo proste w swojej budowie i działaniu, łatwo jest je symulować za pomocą komputera, a specjalistyczne układy scalone mogą zawierać setki takich elementów w jednej kości.

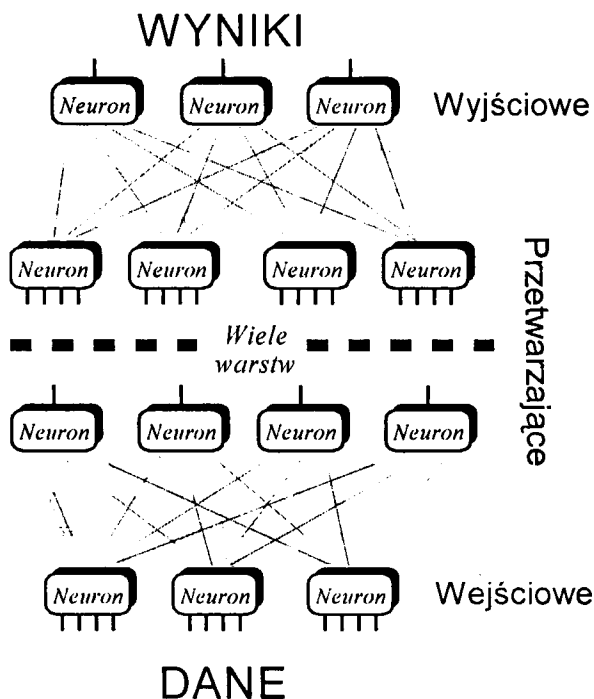
Sztuczny neuron, budulec sieci neuronowej, jest układem mającym wiele wejść (czyli odbierającym i przetwarzającym wiele sygnałów wejściowych z różnych źródeł) i jedno wyjście. Podobnie jak komórka nerwowa (patrz rys. 10.1.).



Rysunek 10.1. Schemat sztucznego neuronu.

Przetwarzanie informacji podanych na wejściu na informację na wyjściu zależy od współczynników nazywanych wagami (oznaczonych na rysunku jako w_1 , w_2 , w_3 , ... w_N), które zmieniają się w procesie uczenia.

Sztuczne neurony łączą się ze sobą w różny sposób, tworząc sieci. Przykład wielowarstwowej struktury sieci neuronowej typu MLP (MultiLayer Perceptron) zobrazowano na rysunku (patrz rys. 10.2.). Sieć taka ma regularną, warstwową, strukturę. Wyróżniamy w niej: neurony wejściowe, które odbierają sygnały z zewnątrz, neurony przetwarzające informację i neurony wyprowadzające końcowy wynik.



Rysunek 10.2. Sieć neuronowa.

Uczenie sieci neuronowej polega na tym, że przedstawia się jej przykłady zadań, dla których znamy poprawne odpowiedzi. Sieć najpierw sama rozwiązuje zadanie, potem porównuje swój wynik z poprawnym rozwiązaniem podanym przez nas. Na początku procesu uczenia stwierdza, że występuje pewien błąd. Wówczas sieć na podstawie porównania obu rozwiązań tak zmienia wartości współczynników („wag”) we wszystkich neuronach, żeby ten błąd zmniejszyć.

Gdy umiemy reagować w określony sposób na określone zbiory danych, można kazać sieci neuronowej rozwiązać zadanie podobne do tych z procesu uczenia. Sieć uogólnia posiadane wiadomości na przypadki, z którymi nigdy się nie zetknęła. Użytkownik może douczać sieć. Jednak trzeba uważać, aby nie „przeuczyć” sieci. Nadmiar treningu jest dla sieci szkodliwy.

Oto kilka przykładów zastosowania sieci neuronowych, przytoczonych za [1998b]:

1. NASA wykorzystuje sieci neuronowe do sterowania ramieniem robota, którego zadaniem jest chwywanie przedmiotów znajdujących się w dowolnym położeniu, dotyczy to między innymi manipulatora działającego w ładowni promów kosmicznych, gdzie w warunkach nieważkości szczególnie trudno jest utrzymać manipulowane obiekty w stałym położeniu, a pewność chwytu jest uzależniona od stopnia zgodności orientacji przestrzennej osi przedmiotu i elementów chwytaka.
2. Również do zadania sterowania ramienia robota desygnowana jest sieć neuronowa zbudowana przez uczonych z New York University Medical Center. Zastosowano sieci jako alternatywę dla złożonych tensorowych obliczeń dotyczących parametrów ruchu ramienia robota w celu znacznego przyspieszenia działania systemu sterowania, który dzięki zastosowaniu sieci może działać w czasie rzeczywistym, podczas gdy zwykłe algorytmy realizowane na tych samych komputerach nie są w stanie osiągnąć wystarczającej szybkości przy wyznaczaniu wszystkich potrzebnych współrzędnych, przyspieszeń i wielkości wymaganych sił i momentów obrotowych.
3. Firma Generat Dynamics opracowała dla potrzeb US Navy oparty na sieciach neuronowych system klasyfikujący i rozpoznający sygnały sonarowe. System ten pozwala rozpoznawać szумы charakterystyczne dla napędów różnych typów statków i okrętów. Udało się tak dokładnie wytrenować sieć, że jest ona zdolna do rozróżniania szumów różnych okrętów tego samego typu, co pozwala na identyfikację nazwy jednostki, a także potrafi rozpoznawać za pomocą hydrofonów niektóre dźwięki nadwodne – na przykład szum wytwarzany przez helikopter unoszący się nad powierzchnią oceanu.
4. Podobne osiągnięcia, ale w zakresie identyfikacji samolotów, uzyskali naukowcy z University of Pensylwania współpracujący z firmą TRW. Zbudowana przez nich sieć neuronowa potrafi rozpoznawać samoloty z taką dokładnością, że podstawą do bezbłędnej identyfikacji może być detal wielkości 18 cali zaobserwowany z odległości 50 mil.

5. W szpitalu Anderson Memorial Hospital w Południowej Karolinie sieci wykorzystano do realizacji procesu optymalizacji leczenia, zyskując (jak podają publikacje) miliony dolarów oszczędności i ratując życie kilkudziesięciu ludzi. Niestety, bliższe dane na temat tego ciekawego przedsięwzięcia nie są znane.
6. Wielki producent raket, firma General Devices Space Systems Division użyła sieci neuronowych do sterowania pracą 150 zaworów doprowadzających paliwo i tlen do silników rakiety Atlas. Okazało się, że odpowiednio wytrenowana sieć potrafiła w oparciu o dane na temat aktualnych fluktuacji mocy silnika sterować zaworami trafniej niż poprzednio stosowany, kosztowny i zawodny złożony system automatyki oparty na wykorzystaniu danych z setek sensorów.
7. Bardzo znanym zastosowaniem sieci neuronowej jest wykorzystany przez firmę Eaton Corporation układ sterowania pozwalający na wspomaganie kierowcy dużej ciężarówki (pięć osi, osiemnaście kół) przy wykonywaniu niektórych szczególnie trudnych manewrów (cofanie z naczepą).
8. Sieci neuronowe często stosowane są w energetyce – zwłaszcza do prognozowania zapotrzebowania na moc elektryczną. Jedno z bardziej znanych wdrożeń w tym zakresie dotyczy elektrowni BC Hydro w Vancouver, szczególnie trudnej do sterowania, ponieważ jej cztery turbiny mają wyraźnie różniące się charakterystyki i zadanie ich włączania i wyłączenia w odpowiedzi na zmienne zapotrzebowanie na moc elektryczną ze strony odbiorców energii jest wyjątkowo złożone i niewdzięczne.

1 1. INTERNET I WWW

Jak zauważono w rozdziale poświęconym sieciom komputerowym, WWW jest częścią Internetu i nie ma między nimi równoważności. W tym rozdziale zostaną omówione podstawowe funkcjonalności Internetu oraz strony WWW zwane niezbyt słusznie stronami internetowymi.

1 1.1. TROCHĘ HISTORII

Pierwszą siecią komputerową była ARPANET i została nazwana „Matką sieci Internet”. W 1969 r. ARPA (Advanced Research Project Agency – Agencja ds. Badań Perspektywicznych) sfinansowała ARPANET, eksperymentalną sieć w dziedzinie łączności z wykorzystaniem tzw. komunikacji pakietów. ARPA była główną agencją Departamentu Obrony (Department of Defense) USA, przekształconą później w DARPA (Defense Advanced Research Projects Agency). DARPA stworzyła dla naukowców połączenie między odległymi centrami komputerowymi, umożliwiając im wzajemne korzystanie ze sprzętu i oprogramowania, przede wszystkim z baz danych, pamięci dyskowej i możliwości obliczeniowych komputerów. To połączenie między sieciami nazwano DARPA Internet, a później skrócono nazwę do Internet.

Pod koniec lat 70. XX w., początkowo na potrzeby uniwersytetów, a potem organizacji komercyjnych, powstały zdecentralizowane, ogólnie dostępne sieci takie jak UUCP łącząca komputery UNIX-owe oraz USENET (User's Network). Na początku lat 80. XX w. udostępniono środowiskom naukowym i akademickim w USA sieci komputerowe Computer Science Network (CSNET) i BITNET. Z początku nie były one częściami sieci Internet, później jednak utworzono specjalne połączenia umożliwiające wymianę danych pomiędzy użytkownikami sieci.

W 1986 r. stworzono, ważną z punktu widzenia historii Internetu, sieć NSFNET (National Science Foundation Network – Sieć Krajowej Fundacji Naukowej) umożliwiającą połączenie naukowców z całych Stanów Zjednoczonych z pięcioma centrami superkomputerowymi.

Służąca do celów naukowych Sieć NSFNET zaczęła stopniowo zastępować ARPANET. W marcu 1990 r. sieć ARPANET została uroczyście odłączona (i rozmontowana). Ponieważ wielu użytkowników sieci CSNET było podłączonych do sieci NSFNET, więc w roku 1991 sieć CSNET również przestała istnieć.

11.2. KOMUNIKACJA W SIECI

W rozdziale o sieciach komputerowych wspomniane zostały standardy komunikacji obowiązujące w sieciach komputerowych – w szczególności model OSI oraz protokół TCP/IP (Transmission Control Protocol/Internet Protocol). Właśnie TCP/IP jest podstawowym standardem komunikacji w rozległych sieciach komputerowych, a tym samym Internetu. Można go nazwać „językiem Internetu”. Protokół TCP/IP powstał w roku 1970 w ramach działań DARPA zmierzających do zapewnienia współdziałania między sieciami (ang. *internetworking*). W ramach eksperymentu przeprowadzono próby łączenia różnych typów sieci i systemów komputerowych. Opracowano i udostępniono nieodpłatnie dla komputerów wykorzystujących system operacyjny UNIX w wersji Berkeley Software Distribution (BSD). Protokołu TCP/IP użyto po raz pierwszy w sieci ARPANET w 1983 roku. Nie jest on chroniony prawami patentowymi i jedynym wymaganiem dla producentów urządzeń sieciowych i komputerów jest spełnienie wymagań modelu OSI, czyli norm ISO. Prace nad TCP/IP zostały sfinansowane z funduszy rządu USA.

Teraz warto poświęcić chwilę uwagi działaniu protokołu. Podczas przesyłania informacji z jednego komputera do drugiego dzielona jest ona na fragmenty zwane pakietami. Każdy pakiet zawiera dane adresowe i przesyłaną informację. Adresy określają, skąd wysłany został pakiet i dokąd ma dotrzeć. Pakiety mogą być wysyłane niezależnie, a ze względu na dużą liczbę połączeń w sieci ich drogi do celu mogą być różne. Nie jest ważna kolejność, w jakiej dotrą do celu, gdyż każdy pakiet zwiiera dane o miejscu, w którym przesłany fragment informacji ma się znaleźć. O wyborze drogi pakietu decydują trasy, o których wspomniano w rozdziale o sieciach.

Sieć Internet nie jest zbudowana według jakiegoś konkretnego modelu, lecz istnieje w niej hierarchia, która wytworzyła się w miarę rozwoju sieci. Szybkie sieci łączące miasta, kraje i kontynenty nazwano sieciami szkieletowymi. Do nich z kolei podłączone są sieci mniejsze, np. MAN i lokalne.

11.3. ADRESY I NAZWY

Do identyfikacji komputera służy adres. Jego konstrukcja powstała w ramach tworzenia protokołu TCP/IP. Adres, a dokładniej adres IP, jest czterobajtową liczbą unikalną dla każdego urządzenia w sieci Internet. Adres ten podaje się w postaci czterech liczb oddzielonych kropkami na przykład 165.44.123.77.

Identyfikacja w postaci ciągu liczb jest przyjazna dla komputerów, lecz nie dla ludzi. Zatem dla ułatwienia z każdym adresem liczbowym skojarzony jest również unikalny identyfikator zwany nazwą. Poszczególne elementy nazwy też są oddzielone kropkami, na przykład poczta.firma.com. Nazwy i adresy komputerów są przechowywane w sieci w specjalnych komputerach w węzłach sieci zwanych serwerami nazw (ang. *name servers*). Nazwy komputerów i innych urządzeń są nadawane w sieciach lokalnych i przekazywane do serwerów nazw. Nazwy tworzone są w System Nazw Domen (Domain Name System), czyli DNS. Skrót DNS oznacza zarazem ogólnosiwiatowy system rozproszonych baz danych zawierających skojarzenie adresu (liczb) z nazwą.

Nazwy skonstruowane są w sposób systematyczny, poszczególne elementy nazwy oddzielone są kropkami, a hierarchię należy odczytywać od końca. Ostatni dwuliterowy człon jest oznaczeniem kraju, w którym znajduje się domena. Na przykład „pl” oznacza Polskę, „ca” – Kanadę, „de” – Niemcy, „uk” – Wielką Brytanię, „us” – Stany Zjednoczone. Ze względów historycznych, ponieważ Internet powstał właśnie w Stanach Zjednoczonych, opuszczane jest dla tego państwa oznaczenie „us”. Kolejny trzyliterowy człon oznacza charakter działalności:

EDU – edukacyjny – *education*,
COM – komercyjny – *commercial*,
GOV – rządowy – *government*,
MIL – wojskowy – *military*,
ORG – organizacja – *organization*,
NET – sieciowy – *networks*.

Ta część nie jest obowiązkowa. Następnym elementem jest nazwa firmy i ta część podlega ochronie. Na przykład IBM ma domenę *ibm.com*. Z tej nazwy widać, że firma IBM prowadzi działalność komercyjną i jej domena jest zarejestrowana w USA. Do kompletności brakuje jeszcze jednego członu, którym najczęściej jest *www*, lecz tu może być np. nazwa działu lub inny wyróżnik określający charakter poddomeny w ramach firmy. Może to być wyraz *marketing* oznaczający dział sprzedaży lub *poczta* oznaczający serwer poczty elektronicznej.

11.4. ZASOBY SIECI INTERNET

Sieć Internet to nie tylko połączenia pomiędzy komputerami użytkowników, ale przede wszystkim zasoby, czyli to wszystko, co może być przydatne i uzyskane za pośrednictwem sieci. Zasobami są nie tylko dokumenty, obrazy, dźwięki, filmy, ale także superkomputery, na których możemy wykonywać obliczenia, komputery z bazami danych oraz inne urządzenia, takie jak np. drukarki. Zasobami sieci Internet mogą być także ludzie, np. konsultanci firm dostępni poprzez pocztę elektroniczną, inni użytkownicy, z którymi mamy kontakt przez listy dyskusyjne.

11.5. DOSTĘP DO ZASOBÓW SIECI INTERNET

Dostęp do zasobów uzyskujemy po podłączeniu się do sieci. W pracy jest to na ogół sieć lokalna, a w domu łączymy się przez np. linię telefoniczną i usługę dostarczaną przez operatora telekomunikacyjnego. Telekomunikacja Polska S.A. oferuje np. usługę pod nazwą Neostrada.

Istnieją trzy główne sposoby dostępu do zasobów sieci: poczta elektroniczna, przesyłanie plików, zdalne rejestrowanie się.

Korzystanie z poczty elektronicznej, podobnie jak z poczty tradycyjnej, wymaga posiadania adresu, pod który możemy otrzymywać listy. Adres poczty elektronicznej składa się z dwóch podstawowych części: *opis użytkownika w miejscu przeznaczenia* oraz *opis miejsca przeznaczenia*. Obie części rozdzielone są symbolem @.

Proszę zwrócić uwagę, że symbol @ oznacza AT (na, w), a nie „matkę”.

Opis użytkownika (ang. *username*) to nazwa naszej skrzynki pocztowej (ang. *mailbox*).

Opis miejsca przeznaczenia to nazwa naszego serwera pocztowego, coś w rodzaju urzędu pocztowego, do którego przychodzą nasze listy, w tym przypadku elektroniczne. Zatem ogólnie można przyjąć: *uzytkownik@miejsc-przeznaczenia*, np. mój adres poczty elektronicznej to: *maciej.pekala@ksw.edu.pl*.

Proszę zwrócić uwagę, że w adresie poczty elektronicznej nie mogą występować znaki diakrytyczne.

Poczta elektroniczna to jedno z najważniejszych narzędzi komunikacji poprzez Internet. Korzystając z niej, możemy zarówno rozsyłać, jak i otrzymywać wiadomości, pliki z danymi, obrazy, pliki dźwiękowe, filmy oraz oprogramowanie. Oczywiście jest, że istnieją ograniczenia rozmiaru informacji, w szczególności plików multimedialnych, jakie możemy przesyłać tą drogą. Dużym udogodnieniem w ramach poczty elektronicznej jest możliwość rozesłania danego listu do wielu odbiorców równocześnie.

Do przekazywania dużych plików lepiej skorzystać z wyspecjalizowanego narzędzia określanego skrótem FTP (ang. *File Transfer Protocol*). Plik jest najmniejszą porcją informacji dostępną dla użytkownika, identyfikowaną poprzez nazwę i położenie w strukturze katalogów (porównaj rozdział o oprogramowaniu). Plik może być dokumentem zawierającym tekst, może zawierać obraz, muzykę, film. Aby skorzystać z tej możliwości, musimy posiadać odpowiedni program lub połączyć się zdalnie z systemem komputerowym.

Podczas połączenia z systemem komputerowym należy znać polecenia umożliwiające przeglądanie struktury katalogów i wyszukiwanie plików. W systemie klasy UNIX służą do tego celu następujące polecenia:

- **dir** (ang. *directory* – katalog),
- **ls** (ang. *list* – spis),
- **cd** nazwa-katalogu (ang. *change directory* – zmień katalog),
- **cdup** (przejdź do katalogu nadrzędnego),
- **pwd** (ang. *print working directory* – wydrukuj nazwę aktualnego katalogu).

Odnalazłszy plik, możemy wydać polecenie transferu (przesłania go) do naszego komputera. Często to polecenie nazywa się właśnie **ftp**.

Ważna jest umiejętność rozpoznawania rodzaju zawartości pliku. Do tego celu służą występujące po kropce trzy ostatnie litery w nazwie pliku nazywane rozszerzeniem nazwy.

Na przykład:

Komputer	Program	Rozszerzenie nazwy	Przesyłanie (typ)
Mac	MacBinary	.bin	binarne (zakodowany)
Mac	Compact Pro	.cpt	binarne (skompresowany)
Mac	BinHex 4.0	.hqx	ASCII (zakodowany)
Mac	self extracting	.sea	binarne (program)
Mac	Stuffit	.sit	binarne (skompresowany)
PC	ARC, PKPAK	.arc	binarne (skompresowany)
PC	ARJ	.arj	binarne (skompresowany)
PC	LHArc	.lzh	binarne (skompresowany)

PC	PAK	.pak	binarne (skompresowany)
PC	Soundblaster	.voc	binarne (dźwięk)
PC	WAVE	.wav	binarne (dźwięk)
PC	PKZIP/InfoZIP	.zip	binarne (skompresowany)
PC	zoo	.zoo	binarne (skompresowany)
UNIX	compress	.Z	binarne (skompresowany)
UNIX	gzip	.gz	binarne (skompresowany)
UNIX	zip & unzip	.zip	binarne (skompresowany)
UNIX	tar	.tar	binarne (archiwum)
każdy	PostScript	.ps	ASCII (opis stron)
każdy	GIF	.gif	binarne (grafika)
każdy	text	.txt	ASCII (tekst)

W kolumnie komputer „Mac” oznacza komputer firmy Apple Macintosh, „PC” – komputer typu IBM PC, „UNIX” – komputer z systemem operacyjnym typu UNIX.

Trzeci sposób dostępu do sieci to zdalne rejestrowanie się (ang. *remote login*). Przydatny jest w sytuacji, gdy chcemy skorzystać z programów i usług dostępnych na innym komputerze. Jest to interakcyjny sposób na wykonanie np. obliczeń na komputerze o dużej mocy obliczeniowej lub wyszukanie informacji w bazie danych. Do zdalnego zarejestrowania się i uzyskania dostępu do zasobów systemu potrzebny jest identyfikator i hasło. Identyfikator jest nazwą użytkownika i może być identyczny z używanym w adresie poczty elektronicznej.

Obecnie uniwersalnym narzędziem dostępu do Internetu łączącym wszystkie trzy sposoby w jednej aplikacji jest przeglądarka internetowa. Jej podstawowe zadanie to przeglądanie dokumentów hipertekstowych, ale często posiada ona dodatkowe funkcje umożliwiające korzystanie z poczty elektronicznej, przesyłanie plików i zdalne rejestrowanie się. Dla sprawnego wyszukiwania informacji w Internecie korzystamy z wyszukiwarek internetowych takich jak: Yahoo, Google, Altavista, Szukacz, Netsearch.

11.6. JĘZYK HTML

HTML (ang. Hypertext Markup Language) jest standardem określającym postać dokumentów przeznaczonych do prezentacji w Internecie. Tworzenie stron WWW nie jest zadaniem trudnym, wymaga jednak wcześniejszego przygotowania projektu stron, przede wszystkim ich szaty graficznej i treści.

Składnia języka HTML opiera się na znacznikach, czyli specjalnych ciągach znaków ujętych w nawiasy: „<” – początkowy, „>” – końcowy. Ponadto znaczniki na ogół występują parami, przy czym znacznik kończący sekwencję zawiera znak „/” (ukośnik). Wewnątrz stron znajdują się odsyłacze zwane adresami URL, o których dalej.

W strukturze każdej strony WWW w języku HTML występują trzy pary znaczników w następującym układzie:


```

<html>
<head>
...   sekcja nagłówka
</head>
<body>
...   sekcja treści
</body>
</html>

```

Znacznik **<html>** otwiera stronę, a **</html>** zamyka. Wszystko, co jest poza tymi znacznikami, jest ignorowane przez przeglądarkę. Tekst umieszczony między znacznikami **<head>** i **</head>** zawiera dane o stronie, autorze oraz informacje dla przeglądarki.

Ciąg znaków: **<title>** Tytuł strony internetowej **</title>** powoduje umieszczenie tytułu strony na pasku tytułowym przeglądarki.

Prawidłowe wyświetlenie znaków diakrytycznych przez przeglądarkę uzyskamy, umieszczając w sekcji nagłówka (head) następujący ciąg znaków:

```
<meta http - equiv = „content - type” content = “ text/html; charset = iso-8859-2”>
```

Chcąc podpisać swoją stronę, powinniśmy użyć następującej sekwencji znaczników i tekstu: **<meta content = „imię i nazwisko” name = author >** .

To, o czym jest nasza strona, oraz słowa kluczowe umieścimy, korzystając z następujących ciągów znaczników i tekstu:

```
<meta content = „O czym jest strona” name = description >
```

```
<meta content = „Co znajduje się na stronie” name = keywords >.
```

W sekcji nagłówka umieszczamy jeszcze inne informacje, o których później.

Teraz przejdziemy do omówienia znaczników występujących w sekcji treści. Różny rozmiar tekstu uzyskamy, korzystając ze znaczników nagłówka. Mamy 6 poziomów. W poniższym przykładzie wykorzystano kilka.

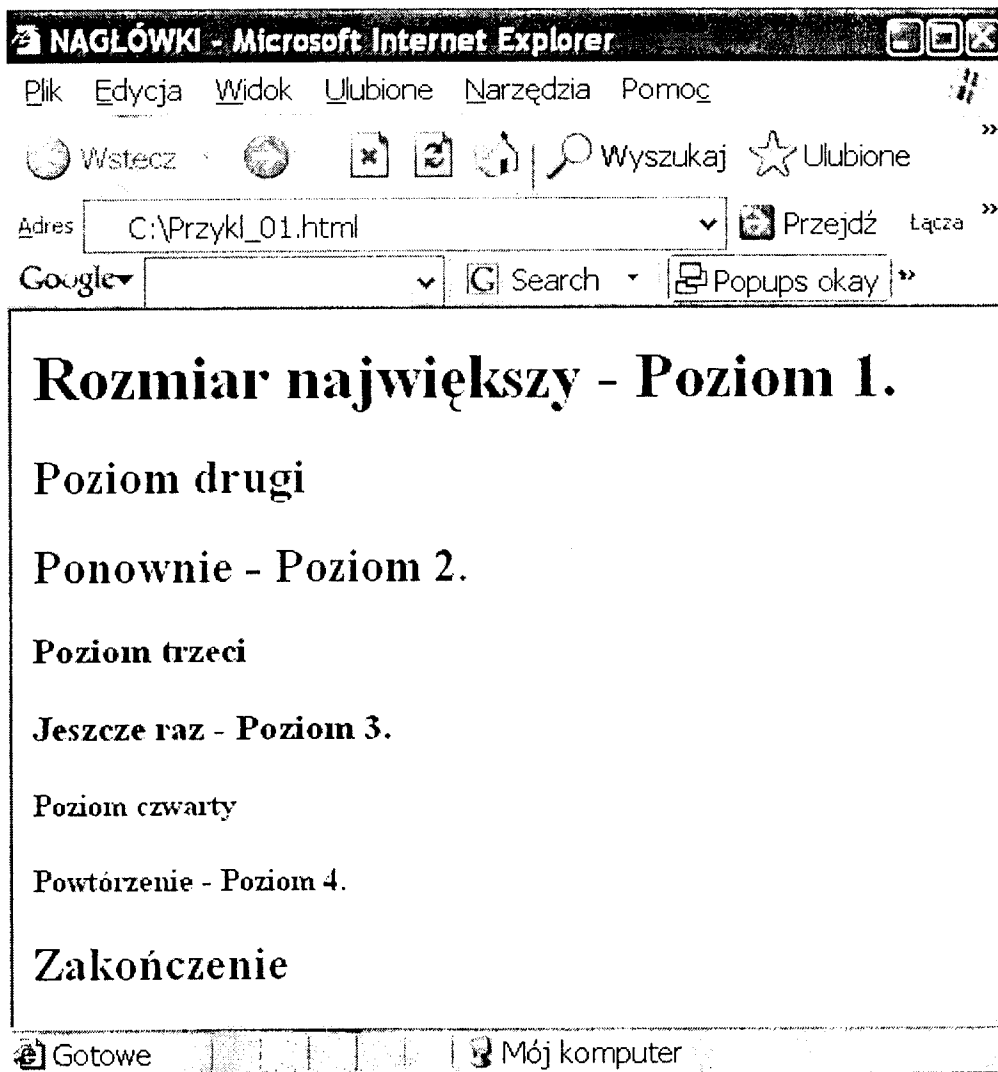
PRZYKŁAD 1.

```

<html>
<title>NAGŁÓWKI</title>
<body>
<h1>Rozmiar największy – Poziom 1.</h1>
<h2>Poziom drugi</h2>
<h2>Ponownie – Poziom 2.</h2>
<h3>Poziom trzeci</h3>
<h3>Jeszcze raz – Poziom 3.</h3>
<h4>Poziom czwarty</h4>
<h4>Powtórzenie – Poziom 4.</h4>
<h2>Zakończenie</h2>
</body>
</html>

```

Efekt uzyskany na ekranie przedstawia rysunek 11.1.



Rysunek 11.1. Obraz ekranu dla Przykładu 1.

Formatowanie tekstu odbywa się odmiennie niż w redaktorze tekstu. Dodatkowe odstępy, zmiana linii oraz akapit muszą być wskazane za pomocą odpowiednich znaczników, a nie znaków wpisanym z klawiatury. Dla zaznaczenia akapitu musimy użyć pary znaczników `<p>` i `</p.>`. Aby uzyskać wcięcie na początku akapitu, musimy użyć znacznika o konstrukcji ` `.

Dla wyrównania tekstu na marginesach lub wyśrodkowania musimy dla danego akapitu użyć atrybutu `align` i nadać mu odpowiednią wartość `center`, `right` albo `left`. Wymuszoną zmianę linii uzyskamy przez znacznik `
`. Dla zwiększenia czytelności rysunku dodano do kodu strony znaczniki nagłówka poziomu drugiego.

Kolejny element formatowania tekstu to krój, rozmiar i kolor czcionki oraz wyróżnienie tekstu. Aby określić krój i rozmiar czcionki, musimy użyć pary znaczników `` i ``. Krój czcionki określamy poprzez atrybut `face`, kolor poprzez atrybut `color`, a rozmiar poprzez `size`. Kolor możemy określić za pomocą nazwy w języku angielskim lub za pomocą kodu szesnastkowego. Nazwy i kody kolorów można znaleźć zarówno w literaturze, jak i w Internecie. Do wyróżniania tekstu mamy następujące pary znaczników: pogrubienie (wyfłuszczenie) `` i ``, kursywa `<i>` i `</i>` oraz podkreślenie `<u>` i `</u>`. Możemy oczywiście używać równocześnie wszystkich trzech znaczników wyróżnienia.

PRZYKŁAD 3.

```

<html>
<title> FORMATOWANIE </title>
<body>
<p>
<font face="Times" color="red" size="8"> <b>Oto czerwona pogrubiona czcionka
Times, rozmiar 8.</b></font>
</p>
<p>
<font face="Arial" color="green" size="7"> <i>Oto zielona kursywa dla czcionki Arial,
rozmiar 7.</i></font>
</p>
<p>
<font face="Courier" color="blue" size="5"> <b>Oto niebieska pogrubiona czcionka
Courier, rozmiar 5.</b></font>
</p>
<p>
<font face="Times" color="black" size="6"> <b>< i><u> Ten tekst jest napisany
pogrubioną, podkreśloną kursywą. </u></i></b></font>
</p>
</body >
</html>

```

W rozdziale o multimediami omówiono hiperłącze. Teraz pokażę, jak zrealizować je w dokumencie hipertekstowym, czyli na stronie internetowej. W języku HTML służy temu para znaczników `<a>` i `` oraz atrybut `href`. Składnia polecenia wygląda tak: ` opis odnośnika`.

Na przykład: ` Krakowska Szkoła Wyższa`

Między znacznikami `<a>` i `` można wstawić dowolny obiekt, w szczególności mały obrazek.

W języku HTML dostępne są trzy struktury listowe: lista wypunktowana, lista numeryczna, lista definicji.

Lista wypunktowana określona jest za pomocą znaczników `` i `` dla całej listy oraz `` i `` dla jej elementów. Domyślnym znakiem wypunktowania jest kropka, jeżeli chcemy zmienić znak, powinniśmy użyć atrybutu **type** i wstawić jedną z wartości:

disc – czarna kropka, znak domyślny,

circle – kółko (puste w środku),

square – kwadrat.

Z kolei lista numeryczna definiowana jest z wykorzystaniem znaczników `` i `` oraz `` i `` dla pozycji listy. Tu atrybut **type** pozwala określić sposób numerowania. Jego wartości są następujące:

1 – liczby, numeracja domyślna,

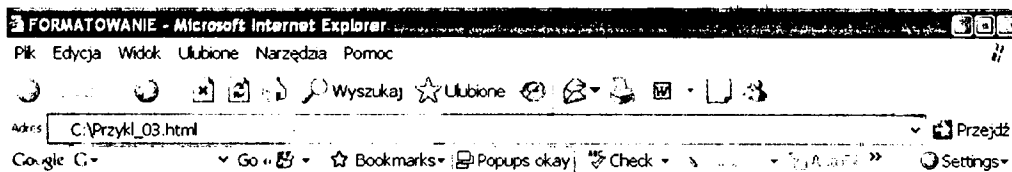
a – małe litery alfabetu,

A – wielkie litery alfabetu,

i – liczby rzymskie pisane małymi literami,

I – liczby rzymskie pisane wielkimi literami.

Ostatnia struktura to lista definicji. Zbudowana jest ona z trzech par znaczników: `<dl>` i `</dl>` ograniczających całą listę, `<dt>` i `</dt>` wyznaczających definiowany termin oraz `<dd>` i `</dd>` ograniczających treść definicji.



Oto czerwona pogrubiona czcionka Times, rozmiar 8.

Oto zielona kursywa dla czcionki Arial, rozmiar 7.

Oto niebieska pogrubiona czcionka Courier, rozmiar 5.

Ten tekst jest napisany pogrubioną, podkreśloną kursywą.

PRZYKŁAD 4.

```

<html>
<title>LISTY</title>
<body>

<h3>
<h2>Lista wypunktowana:</h2>
<ul type=square>
<li>Pozycja nr 1.</li>
<li>Pozycja nr 2.</li>
<li>Pozycja nr 3.</li>
</ul>

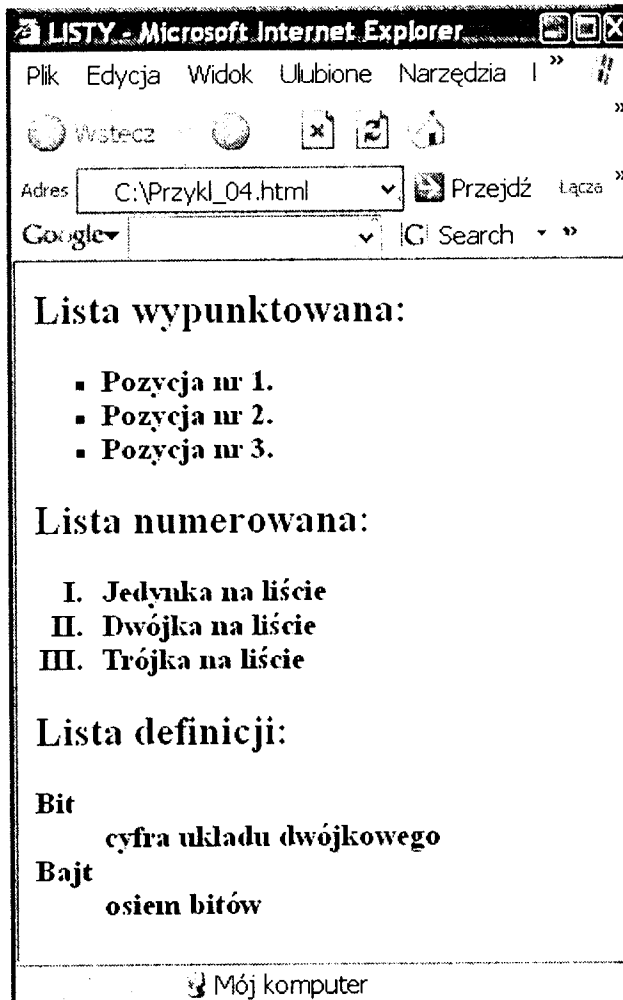
<h2>Lista numerowana:</h2>
<ol type=I>
<li>Jedynka na liście</li>
<li>Dwójka na liście</li>
<li>Trójka na liście</li>
</ol>

<h2>Lista definicji:</h2>
<dl>

<dt>Bit</dt>
<dd>cyfra układu dwójkowego</dd>

<dt>Bajt</dt>
<dd>osiem bitów</dd>
</dl>
</h3>
</body>
</html>

```



Rysunek 11.4. Obraz ekranu dla Przykładu 4.

Nieodłącznym elementem stron internetowych są obrazy. Do wstawienia obrazu wewnątrz strony służy znacznik ``. O plikach graficznych mówiliśmy w rozdziale poświęconym multimediom. Atrybut `alt` umożliwia wstawienie tekstu, który będzie wyświetlany w przypadku, gdy z jakichś przyczyn obraz nie może być wyświetlony. W tabeli 11.1 zestawiono wybrane atrybuty, które umożliwiają odpowiednie dopasowanie i ułożenie obrazu na stronie.

Tabela 11.1. Wybrane atrybuty znacznika .

align=	left	obraz będzie wyświetlony przy lewym marginesie, a tekst między obrazem a prawym marginesem
	right	obraz będzie wyświetlony przy prawym marginesie, a tekst między lewym marginesem a obrazem
	top middle baseline bottom	pionowe położenie obrazu względem tekstu, znajdującego się w tym samym wierszu
vspace= liczba_pikseli		odstęp między obrazem a tekstem w pionie
hspace= liczba_pikseli		odstęp między obrazem a tekstem w poziomie
height= liczba_pikseli liczba_%		określenie wysokości bezwzględnej obrazu w pikselach albo w procentach w stosunku do oryginału
width= liczba_pikseli liczba_%		określenie szerokość bezwzględnej obrazu w pikselach albo w procentach w stosunku do oryginału

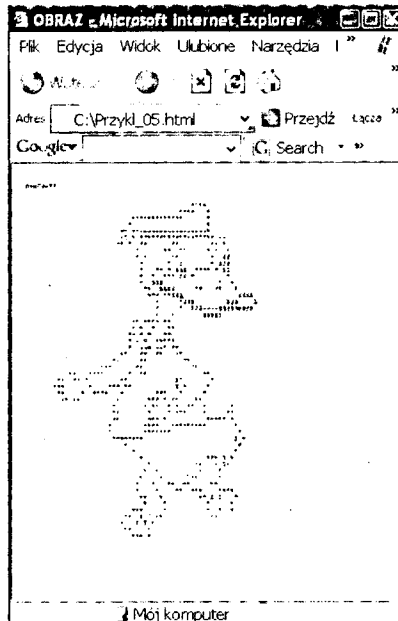
PRZYKŁAD 5.

```

<html>
<title> OBRAZ</title >
<body>

</body>
/html>

```



Rysunek 11.5. Obraz ekranu dla Przykładu 5.

Do budowy stron internetowych przydatne są też tabele. Zawartością tabeli mogą być teksty, obrazy, listy, odnośniki. Do definiowania tabeli służą pary znaczników `<table>` i `</table>`, poszczególne wiersze tabeli uzyskujemy przez pary znaczników `<tr>` i `</tr>`, a komórki w wierszu definiują znaczniki `<td>` i `</td>`, które umieszczamy między znacznikami wiersza. Szerokość obramowania tabeli określimy poprzez atrybut **border**. Domyślnie tabele nie mają obramowań. W tabeli poniżej zebrano niektóre przydatne atrybuty i znaczniki tabel.

Tabela 11.2. Zestawienie wybranych atrybutów dla `<table>`.

Znaczniki	Wybrane atrybuty
Tabela <code><table></code> <code></table></code>	cellspacing = liczba pikseli – wolny obszar między komórkami align = <i>param</i> – sposób wyrównania tabeli, <i>param</i> przyjmuje wartości jak wyżej cellpadding = liczba pikseli – odstęp między krawędziami komórki a jej zawartością bgcolor = <i>kolor</i> – zmienia tło tabeli, <i>kolor</i> przyjmuje wartości jak wyżej
Wiersz <code><tr></code> <code></tr></code>	align = <i>param</i> – znaczenie jak wyżej bgcolor = <i>kolor</i> – znaczenie jak wyżej bordercolor = <i>kolor</i> – określa kolor obramowania, <i>kolor</i> ma znaczenie jak wyżej
Komórka <code><td></code> <code></td></code>	rowspan = <i>n</i> – komórka będzie przebiegać przez <i>n</i> wierszy colspan = <i>n</i> – komórka będzie przebiegać przez <i>n</i> kolumn width = liczba_pikseli liczba_% – szerokość komórki w pikselach albo procentach względem szerokości tabeli
Komórka nagłówka <code><th></code> <code></th></code>	Wpisany do takiej komórki tekst jest domyślnie pogrubiony i wyśrodkowany. Atrybuty jak przy <code><td></code>
Stopka tabeli <code><tfoot></code> <code></tfoot></code>	align = <i>param</i> – znaczenie jak wyżej
Nagłówek tabeli <code><thead></code> <code></thead></code>	align = <i>param</i> – znaczenie jak wyżej
Główna treść tabeli <code><tbody></code> <code></tbody></code>	align = <i>param</i> – znaczenie jak wyżej

PRZYKŁAD 6.

```

<html>
<title>TABELE</title>
<body>
<table border=2>
<tr>
<th>Nagłówek k_1</th>
<th>Nagłówek k_2</th>
</tr>
<tr>
<td><h3>Element w_1, k_1 .....</h3></td>
<td><h3>Element w_1, k_2 .....</h3></td>

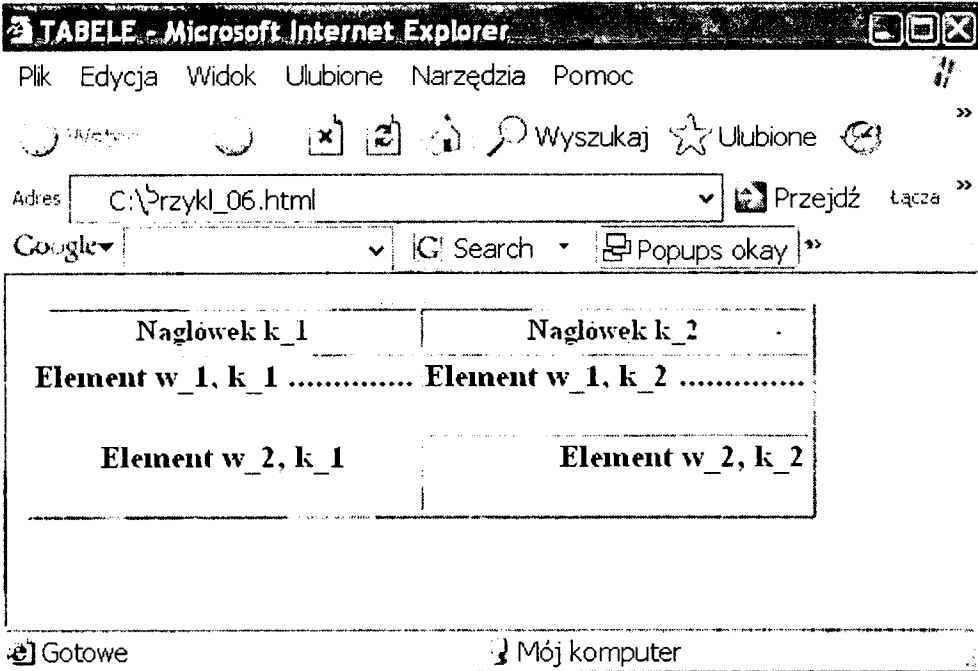
```



```

</tr>
<tr>
<td align="center"><h3>Element w_2, k_1 </h3></td>
<td align="right"><h3>Element w_2, k_2 </h3></td>
</tr>
</table>
</body>
</html>

```



Rysunek 11.6. Obraz ekranu dla Przykładu 6.

Efektowny układ strony możemy uzyskać, korzystając z ramek. Do definiowania układu ramek służy para znaczników `<frameset>` i `</frameset>` o atrybutach `cols` i `rows`. Atrybut `col` określa liczbę ramek pionowych, na które zostanie podzielone okno przeglądarki, natomiast atrybut `rows` określa liczbę ramek poziomych w oknie przeglądarki.

Poszczególne ramki definiujemy, korzystając ze znaczników `<frame>` i `</frame>`. Niektóre atrybuty ramki to:

- `marginheight` = „liczba_pikseli” – określa wysokość marginesu,
- `marginwidth` = „liczba_pikseli” – określa szerokość marginesu,
- `frameborder` = „0 | 1” – obramowanie ramki ma być wyświetlane 1, nie wyświetlane 0,
- `scrolling` = „yes | no | auto” – sposób przewijania ramki,
- `name` = „nazwa_ramki” – umożliwia podanie nazwy ramki,
- `src` = „url” – nazwa i miejsce pliku, który zostanie wyświetlony w ramce, np. obrazka.

Pamiętajmy o wcześniejszym przygotowaniu odpowiednich plików.

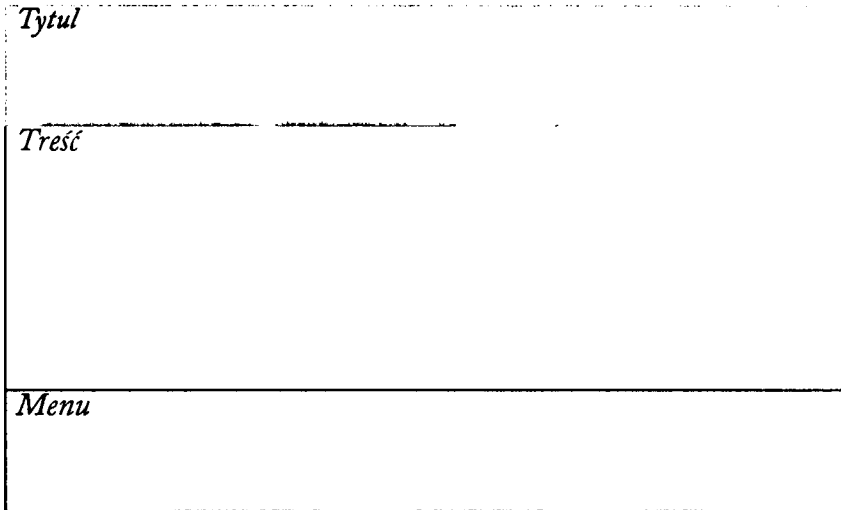
Uwaga: znak „|” w powyższym opisie rozdziela wartości atrybutu do wyboru.

Poniżej znajdują się dwa przykłady stron zbudowanych z wykorzystaniem ramek.

Pierwszy składa się wyłącznie z ramek poziomych, a drugi jest układem mieszanym.

Ramki w wersji poziomej: *Tytuł – Treść – Menu*

Najpierw projekt:



PRZYKŁAD 7.

```
<html>
<head>
<title>MOJE_RAMKI</title>
</head>
<frameset rows="10%,*,10%" frameborder="1" border="5" framespacing="5">
<frame name="Tytuł" SRC="tytul.html" marginheight="0" marginwidth="0" frame-
border="0" noresize>
<frame name="Tresc" SRC="kaczor_bc.jpg" marginheight="0" marginwidth="0" frame-
border="0" border="0">
<frame name="Menu" SRC="menu.html" marginheight="0" marginwidth="0" frame-
border="0" noresize>
</frameset>
<noframes>
Twoja przeglądarka nie obsługuje ramek. <br>
Aby obejrzeć tę stronę, musisz mieć zainstalowaną przeglądarkę obsługującą ramki.
<br>
Np. Netscape Navigator 3.x lub Internet Explorer 3.x
</noframes>
</html>
```


Ramki w wersji mieszanej: *Tytuł* – *Treść* – w pionie – *Menu* – z lewej

Tytuł	
Menu	Treść

PRZYKŁAD 8.

```

<html>
<head>
<title>MOJE_RAMKI</title>
</head>
<frameset rows="10%,*" frameborder="1" border="2">
<frame name="Tytuł" src="tytul.htm" marginheight="0"
marginwidth="0" frameborder="1" border="2">
<frameset cols="15%,*" frameborder="1" border="2">
<frame name="Menu" src="menu2.htm" marginheight="0"
marginwidth="0" frameborder="1" border="2">
<frame name="Tresc" src="kaczor_bc.jpg" marginheight="0"
marginwidth="0" frameborder="1" border="2" noresize>
</frameset>
</frameset>
<noframes>
<body>
<p>Twoja przeglądarka nie obsługuje ramek. <br>
Aby obejrzeć tę stronę, musisz mieć zainstalowaną przeglądarkę obsługującą ramki.
<br>
Np. Netscape Navigator 3.x lub Internet Explorer 3.x </p>
</body>
</noframes>
</frameset>
</html>

```

Pliki potrzebne do wyświetlenia w ramkach:

```
<html>
```

```
<title> TYTUŁ </title>
```

```
<body>
```

```
<font align="center" face="Times" color="red" size="8"> <B><p align="center"> MOJE  
OBRAZKI </p></b></font>
```

```
</body >
```

```
</html>
```

```
<html>
```

```
<title> MENU_2 </title>
```

```
<body>
```

```
<p ><br><br><br>
```

```
<font face="Arial" color="black" size="5"> <B>Pierwszy </b></font><br><br><br>
```

```
<font face="Arial" color="black" size="5"> <B>Poprzedni </b></font><br><br><br>
```

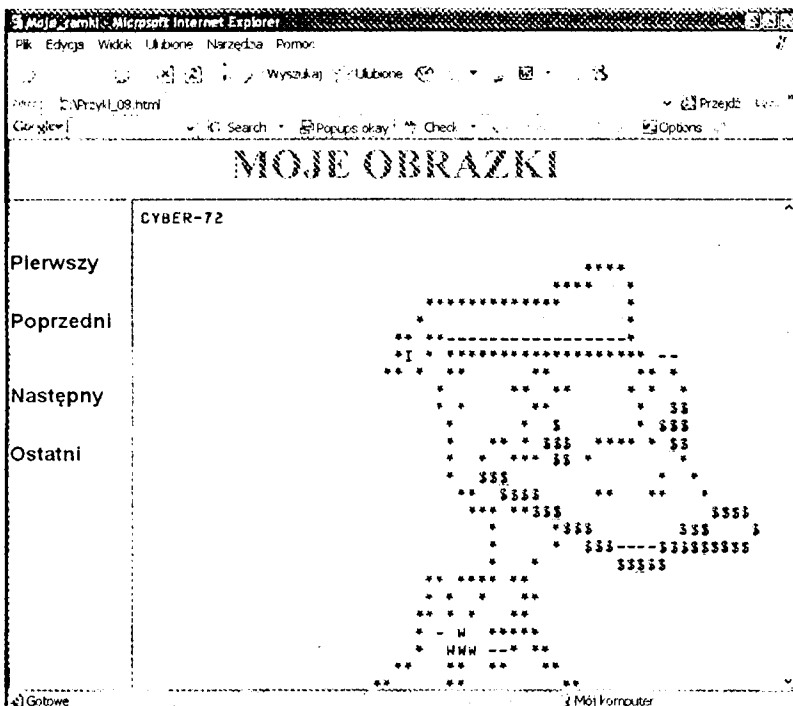
```
<font face="Arial" color="black" size="5"> <B>Następny</b></font><br><br><br>
```

```
<font face="Arial" color="black" size="5"> <B>Ostatni </b></font>
```

```
</p>
```

```
</body >
```

```
</html>
```

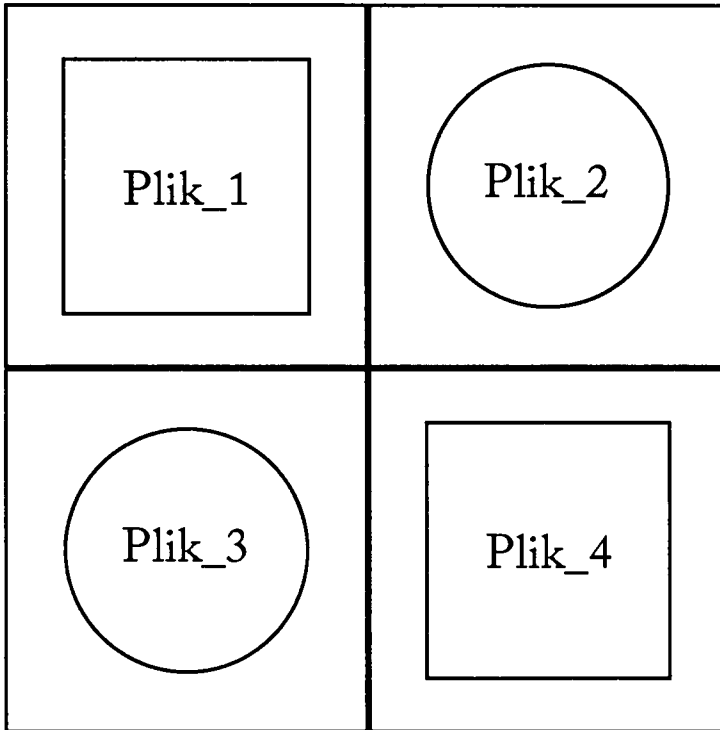


Rysunek 11.8. Obraz ekranu dla Przykładu 8.

Na zakończenie tego krótkiego zarysu języka HTML najbardziej efektywna, ale najtrudniejsza w przygotowaniu mapa graficzna.

Najpierw przygotowujemy obrazek i planujemy obszary aktywne. Dla każdego obszaru aktywnego definiujemy odpowiadający mu plik np. w języku HTML lub obrazek.

Poniżej przedstawiam projekt mapy:



Mapa planowana jest na obrazku o nazwie *menuobr.gif*.

Mamy dwa obszary prostokątne i dwa koliste. Punkt o współrzędnych (0,0) położony jest w lewym górnym rogu. Najpierw podajemy wartość współrzędnej poziomej, a następnie pionowej. Zakładamy, że obrazek jest kwadratem o rozmiarach 400 x 400 pikseli. Kwadrat ten podzielony jest na cztery równe kwadraty, w których zaplanowano obszary reagujące na kliknięcie lewym klawiszem myszy. Dla *Plik_1* kwadrat o współrzędnych przeciwległych wierzchołków (50,50) i (150,150), dla *Plik_2* koło o środku (300,100) i promieniu 50 pikseli, dla *Plik_3* koło o środku (100,300) i promieniu 50 pikseli, dla *Plik_4* kwadrat o współrzędnych przeciwległych wierzchołków (250,250) i (350,350)

Oto odpowiedni kod w języku HTML. Proszę zwrócić uwagę, że opis mapy znajduje się w sekcji nagłówka, a w sekcji treści jest tylko odwołanie.

PRZYKŁAD 9.

```

<html>
<head>
<title> MAPA GRAFICZNA </title>
</head>
<map name="mapa1">
<area shape="rect" coords="50,50,150,150"
href="plik_1.htm">
<area shape="circ" coords="300,100,50"
href="plik_2.htm">
<area shape="circ" coords="100,300,50"
href="plik_3.htm">
<area shape="rect" coords="250,250,350,350"
href="plik_4.htm">
</map>
<body>

</body>
</html>

```

Poniżej przedstawiam dwa przykłady map graficznych opartych na powyższym schemacie. Najpierw obrazki:

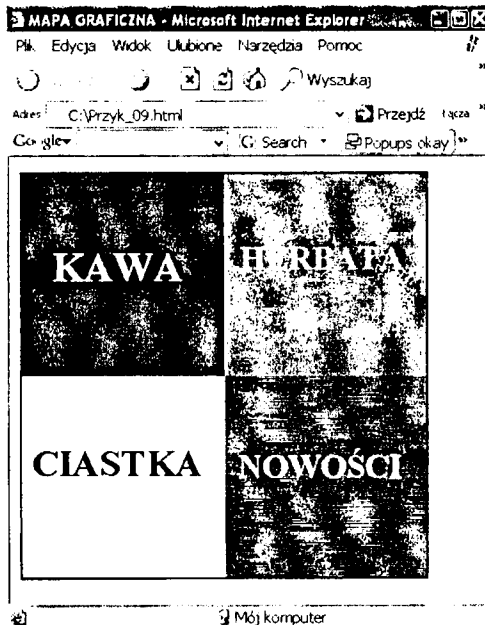


Rysunek 11.9. Obraz *menuobr.gif* do Przykładu 9.

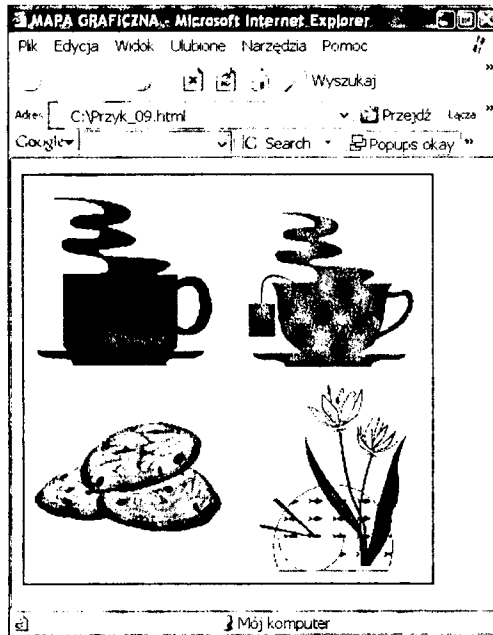


Rysunek 11.10. Obraz *picmenu.gif* do Przykładu 9.

Teraz wygląd strony w przeglądarce:



Rysunek 11.11. Obraz ekranu dla Przykładu 9.



Rysunek 11.12. Obraz ekranu dla Przykładu 9 po zmianie obrazka.

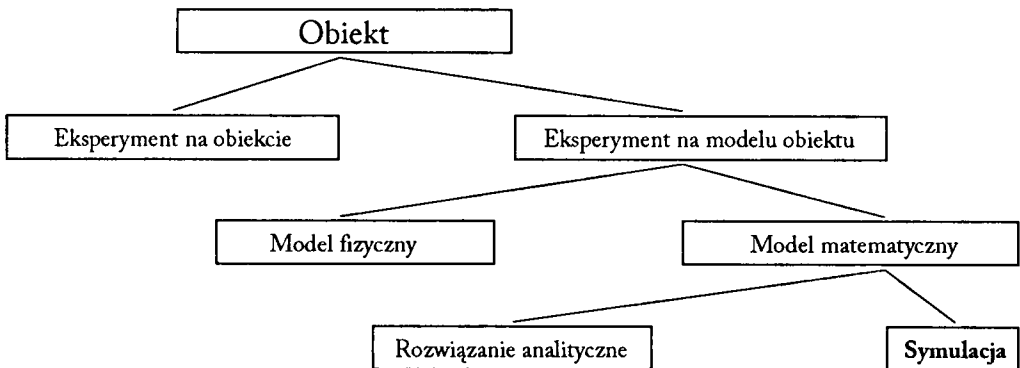
Dalsze studiowanie języka HTML oraz eksperymenty z własnymi stronami WWW pozostawiam Czytelnikowi.

12. SYMULACJA

- [1997] SYMULACJA KOMPUTEROWA, metoda wnioskowania o zachowaniu się obiektów rzeczywistych na podstawie obserwacji działania programów komputerowych symulujących to zachowanie (tzw. modeli symulujących); program symuluje zachowanie się obiektu, gdy zbiór stanów, w których może znajdować się komputer sterowany przez ten program, odpowiada stanom obiektu, i gdy zmiana stanu w komputerze obrazuje zmianę odpowiedniego stanu w obiekcie. Rozróżnia się symulację ciągłą i dyskretną; symulację komputerową ciągłą można realizować ściśle w komputerach analogowych (mech., elektromech. lub pneumatyczne urządzenie, w którym przetwarzana informacja jest przedstawiana w analogowej, tj. ciągłej, postaci), w komputerach cyfrowych – tylko z pewnym przybliżeniem; komputery analogowe mają mniejszą dokładność obliczeń. Symulację komputerową stosuje się np.: gdy bezpośrednie obserwowanie zachowania się obiektu jest niemożliwe (np. przy projektowaniu) lub trudne, gdy przeprowadzenie eksperymentów z obiektem wymaga kosztownych urządzeń, gdy metody mat. nie dają zadowalających wyników itp.; obiektami mogą być urządzenia mech., układy elektron., obiekty astr., grupy społ., systemy ekon., organizmy żywe lub ich populacje, a także programy komputerowe (symulowane przez inne programy), automaty abstrak. i in.; często stosuje się też technikę symulowania komputera (np. dopiero projektowanego) na innym komputerze.
- [1999a] SYMULACJA – technika analityczna, w której analityk obserwuje cechy obiektu, tworząc jego model, a następnie badając zachowanie tego modelu. Np. inżynierowie lotnictwa używają technik symulacji komputerowej do projektowania i testowania (w krótkim czasie) tysięcy alternatywnych modeli samolotów, wypierając w nowoczesnych firmach tunele aerodynamiczne. Symulacja jest również stosowana w edukacji do przeprowadzania wirtualnych eksperymentów naukowych i w biznesie (do wykonywania finansowych analiz wariantowych). Jednakże jak w każdym modelu, symulacja jest tak dobra, jak dobre są przyjęte założenia. Jeśli założenia nie są poprawne, to uzyskany model nie naśladuje dokładnie zachowania się rzeczywistego systemu, który jest symulowany.

Symulacja pozwala nam niższym kosztem uzyskać wiedzę o obiektach, procesach, systemach. Zagadnienie jest bardzo rozległe i ma wiele aspektów. W tym rozdziale zajmiemy się wizualnym aspektem symulacji, w nawiązaniu do komputerowych metod prezentacji. Najpierw kilka uwag ogólnych.

Rozważmy najpierw możliwe warianty metod uzyskiwania wiedzy o obiekcie.



Oczywiście najbardziej kosztowne i ryzykowne są eksperymenty na obiekcie. Przygotowuje się zatem model fizyczny bądź matematyczny. W naszych rozważaniach interesujemy się modelem matematycznym, a tu z kolei symulacją. Symulacja może być realizowana na wiele sposobów, lecz tu zwrócimy uwagę na techniki symulacji z wykorzystaniem komputerów.

Nie ma uniwersalnych modeli symulacyjnych. W zależności od charakteru obiektu musimy dobrać model. Klasyfikację modeli możemy przeprowadzić ze względu na trzy aspekty: ciągłość, dynamikę i determinizm.

Zdarzenia ciągłe i dyskretne

Zdarzenie dyskretne występuje w określonych momentach czasu, niekoniecznie systematycznie. Konstruując model takiego zdarzenia, wyróżniamy przedziały czasowe, w których zdarzenie zachodzi. Przykłady takich zdarzeń: klient wchodzący do sklepu, autobus odjeżdżający z przystanku.

Zdarzenie ciągłe jest to zjawisko, które trwa bez przerwy. Ciągłość nie jest przerywana chwilowymi operacjami, tak jak to jest w przypadku zdarzeń dyskretnych. Przykładami takich zdarzeń są: temperatura powietrza w ciągu dnia, napełnianie baku samochodu paliwem.

W praktyce trudno znaleźć zjawiska całkowicie dyskretne lub całkowicie ciągłe. Model dobiera się według dominującej cechy zjawiska.

Stacyczne i dynamiczne modele symulacyjne

Stacyczny model symulacyjny charakteryzuje się niezależnością od czasu. Jest to model, w którym upływ czasu nie ma wpływu na wynik eksperymentu. Rzut monetą lub kostką jest tu dobrym przykładem. Zjawiska takie modeluje się z wykorzystaniem metod Monte Carlo. Dynamiczny model symulacyjny charakteryzuje się zależnością od czasu. W tym modelu upływ czasu ma istotny wpływ na wynik eksperymentu, a stan obiektu zmienia się wraz z upływem czasu. Przykładem może być jazda samochodem, podczas której droga zależy od czasu i oczywiście prędkości chwilowej.

Deterministyczne i stochastyczne modele symulacyjne

W modelu symulacji deterministycznej nie wykorzystuje się zjawisk losowych. Oznacza to, że przebieg eksperymentu nie podlega prawom rachunku prawdopodobieństwa, nie jest zjawiskiem losowym i daje się przewidzieć.

Model symulacji stochastycznej opiera się na ciągach losowo wygenerowanych wartości, zwanych procesami stochastycznymi. Wartości zmieniają się w sposób losowy i musimy posługiwać się metodami rachunku prawdopodobieństwa. Przykładem takiego zjawiska jest czas, jaki upływa między kolejnymi awariami określonego egzemplarza samochodu danej marki.

Narzędzia symulacyjne

Dobór narzędzia jest jedną z ważniejszych decyzji i zależy od wielu czynników. W szczególności od doświadczenia eksperymentatora, czasu i środków finansowych, które mamy do dyspozycji, możliwości samego narzędzia.

Pierwsza, stosunkowo łatwo dostępna grupa narzędzi to arkusze kalkulacyjne (porównaj rozdział 7). Nie są one na ogół kojarzone z możliwością budowania modeli symulacyjnych. Umożliwiają jednak budowę modeli zarówno deterministycznych, jak i stochastycznych. Przy budowie modeli stochastycznych wykorzystujemy funkcję generatora liczb losowych o nazwie RAND lub podobnej. Często arkusze kalkulacyjne wykorzystuje się do budowania eksperymentów finansowych w zarządzaniu firmą. Dzięki wbudowanym mechanizmom wizualizacji wyników poprzez wykresy uzyskujemy doskonałe narzędzie do szybkich eksperymentów z czytelną prezentacją wyników.

Druga grupa to narzędzia do szybkiej symulacji. Zadaniem tych narzędzi jest określenie elementów krytycznych zjawiska, procesu bądź systemu w krótkim czasie, lecz nie wymaga się od nich dużej dokładności. Na ogół są wąsko specjalizowane.

Trzecia grupa to symulatory. Pozwalają one na budowę modeli specyficznych zagadnień. Na ogół nie wymagają od użytkownika znajomości języka programowania i zapewniają atrakcyjną prezentację wyników w postaci graficznej. Także czas przygotowania eksperymentu jest stosunkowo krótki, choć symulowanie innych problemów niż określone przez producenta nie jest możliwe. Symulatory umożliwiają eksport wyników do innych programów, np. graficznych.

Ostania grupa narzędzi to języki symulacyjne. Można oczywiście przy budowie eksperymentu korzystać z takich języków programowania jak Pascal lub C++, ale lepsze wyniki uzyskamy, korzystając z języków specjalizowanych. Przykładami takich języków są SIMULA i SIMSCRIPT. Zastosowanie języków symulacyjnych pozwala na bardzo szczegółowe badanie obiektu. Wadą jest stosunkowo długi czas przygotowania eksperymentu i konieczna znajomość języka programowania.

Wszystkie narzędzia w sposób bezpośredni bądź pośredni umożliwiają wizualizację przebiegu i wyników symulacji. Otrzymujemy albo szereg pojedynczych obrazów albo całe filmy, często bardzo widowiskowe.

ŹRÓDŁA

(W KOLEJNOŚCI CHRONOLOGICZNEJ)

- [1973] *Mały słownik cybernetyczny*, Warszawa 1973.
- [1975] Turski W.M., *Propedeutyka informatyki*, Warszawa 1975.
- [1976a] *Słownik informatyki polsko-angielsko-rosyjski*, Warszawa 1976.
- [1976b] Martin J., *Dialog człowieka z maszyną cyfrową*, tłum. Cz. Syc, L. Maciejec, T. Sinkiewicz, A. Zakrzewski, Warszawa 1976.
- [1977] Löbel G., Müller P., Schmid H., *Leksykon informatyki*, tłum. W. Klepacz, Warszawa 1977.
- [1981] Corge Ch., *Elementy informatyki. Informatyka a myśl ludzka*, tłum. M. Obrębska, Warszawa 1981.
- [1982] *Mały ilustrowany leksykon techniczny*, Warszawa 1982.
- [1987] Madej D., Marasek K., Kuryłowicz K., *Komputery osobiste*, Warszawa 1987.
- [1988] Kleiber M., Szuniewicz R., *Komputer osobisty Typu IBM PC. Możliwości zastosowań profesjonalnych*, Warszawa 1988.
- [1989] Gniłka M., Haschka J., *Istota informatyki*, Warszawa 1989.
- [1992] Gieroń M., *TIFF, PCX, PIC. Formaty plików graficznych*, Tarnów 1992.
- [1994a] LaQuey T., Ryer J.C., *Internet i okolice. Przewodnik po światowych sieciach komputerowych*, tłum. M. Zielińska, Warszawa 1994.
- [1994b] Parker R.C., *Skład komputerowy w minutę*, tłum. J. Mirkowski, Warszawa 1994.
- [1997] *Nowa encyklopedia powszechna PWN*, Warszawa 1997.
- [1998a] Beynon-Davies P., *Systemy baz danych*, tłum. L. Banachowski, M. Banachowski, Warszawa 1998.
- [1998b] Tadeusiewicz R., *Elementarne wprowadzenie do techniki sieci neuronowych z przykładowymi programami*, Warszawa 1998.
- [1999a] Pfaffenberger B., *Słownik terminów komputerowych*, tłum. zbiorowe, Warszawa 1999.
- [1999b] *Vademecum. Teleinformatyka*, Warszawa 1999.
- [2000a] *Encyklopedia wiedzy komputerowej*, Warszawa 2000.
- [2000b] *HTML. Leksykon kieszonkowy*, Gliwice 2000.
- [2003] *Słownik informatyczny*, Wrocław 2003.
- [2004] Bradbury A., *Jak zorganizować prezentację*, T. Rzychoń, Gliwice 2004.

STRONY WWW

- [1] http://pl.wikipedia.org/wiki/Strona_główna, 1.10.2006.
- [2] <http://encyklopedia.pwn.pl/>, 1.10.2006.
- [3] <http://sjp.pwn.pl/>, 1.10.2006.

LITERATURA PRZEDMIOTU DO SAMODZIELNEGO STUDIOWANIA

Do CAŁOŚCI:

1. Durka P.J., *Komputer. Internet. Cyfrowa rewolucja*, Warszawa 2000.

Do ROZDZIAŁU 1:

1. Harel D., *Rzecz o istocie informatyki. Algorytmika*, tłum. Z. Weiss, P. Carlson, Warszawa 2001.

Do ROZDZIAŁU 2:

1. Brookshear G.J., *Informatyka w ogólnym zarysie*, tłum. M. Engel, Warszawa 2003.
2. David H., *Komputery – spółka z o.o. Czego komputery naprawdę nie umieją robić*, tłum. A. Schubert, Warszawa 2003.
3. Hyde R., *Profesjonalne programowanie. Część 1. Zrozumieć komputer*, tłum. T. Żmijewski, Gliwice 2005.
4. Mueller S., *Rozbudowa i naprawa komputerów PC*, tłum. P. Pilch, Gliwice 2003.
5. Strona internetowa o historii komputera: <http://historia.adrem.pl/index.php3>, 1.10.2006.

Do ROZDZIAŁU 3:

1. Brown H.T., Danda M., *Microsoft Windows XP w małych sieciach komputerowych*, tłum. J. Machowski, Warszawa 2002.
2. Derfler F.J., *Sieci komputerowe dla każdego*, tłum. D. Boratyn, Gliwice 2001.
3. Gast M.S., *802.11. Sieci bezprzewodowe. Przewodnik encyklopedyczny*, tłum. A. Romanek, W. Ziolo, Gliwice 2003.
4. Habraken J., *ABC sieci komputerowych*, tłum. M. Michalski, W. Ziolo, Gliwice 2002.
5. Hemenway K., Calishain T., *100 sposobów na tworzenie robotów sieciowych*, tłum. T. Żmijewski, Gliwice 2004.
6. Kurose J.F., Ross K.W., *Sieci komputerowe. Od ogółu do szczegółu z internetem w tle*, tłum. P. Pilch, G. Werner, Gliwice 2006.
7. Krysiak K., *Sieci komputerowe*, Gliwice 2005.
8. Kurata D., *Net. Najpilniej strzeżone tajemnice*, Gliwice 2005.
9. Podstawczyński A., *Linux w sieci*, Gliwice 2002.
10. Szeliga M., Nieszwiec S., Bachman R., Kura M., Michalski T., *Sieci P2P Wymiana plików w Internecie*, Gliwice 2004.

Do ROZDZIAŁU 4:

1. *Podstawy użytkowania komputerów*, red. M. Zając, Wilkowiec 2001.
2. Ball B., *Red Hat Linux 7.3. Księga eksperta*, tłum. M. Pasternacki, Gliwice 2002.
3. Golał K., Golał R., *Prawo komputerowe (zagadnienia podstawowe)*, Warszawa 1998.
4. Stencil K., *Systemy operacyjne*, Warszawa 2005.
5. Hohmann L., *Więcej niż architektura oprogramowania*, tłum. P. Koronkiewicz, Gliwice 2006.
6. Marczyński J., *UNIX użytkowanie i administrowanie*, Gliwice 2000.
7. Matlak A., *Prawo autorskie w społeczeństwie informacyjnym*, Kraków 2004.
8. Morimoto R., Noel M., Droubi O., Gardinier K., Neal N., *Windows Server 2003. Księga eksperta*, tłum. A. Jarczyk, T. Miszkiewicz, P. Pilch, Gliwice 2004.
9. Ogletree T.W., *Windows XP PL. Księga eksperta*, tłum. G. Kowalczyk, A. Kula, C. Welsyng, Gliwice 2002.
10. Parker T., *Linux. Księga eksperta*, tłum. K. Olesiejuk, Gliwice 2001.
11. Simpson A., *Windows XP PL. Biblia*, tłum. M. Gołębiowski, P. Pilch, K. Słodownik, Gliwice 2003.

Do rozdziału 9:

1. *Multimedialne i sieciowe Systemy informacyjne. Materiały konferencyjne*, red. Cz. Daniłowicz, Wrocław 2004.
2. Jankowski M., *Elementy grafiki komputerowej*, Warszawa 2006.
3. Oberlan Ł., Zimek R., *ABC grafiki komputerowej*, Gliwice, 2004.
4. Zarzecki K., *ABC Corela 9.0*, Kraków 2001.

Do rozdziału 6:

1. Dec Z., *ABC Worda*, Kraków 2000.
2. Diller A., *LaTeX. Wiersz po wierszu*, tłum. J. Jelowicki, Gliwice 2001.
3. Dziewoński M., *OpenOffice 2.0 PL. Oficjalny podręcznik*, Gliwice 2005.
4. Kwaśny A., *DTP Księga eksperta*, Gliwice 2002.
5. Williams R., *Jak składać tekst? Komputer nie jest maszyną do pisania*, tłum. A. Jarczyk, wyd. 2, Gliwice 2003.
6. Williams R., *Komputerowy skład tekstów. Jak to zrobić?*, tłum. A. Bulandra, Gliwice 2003.

Do rozdziału 7:

1. Bernard V. L., *Microsoft Excel w biznesie i zarządzaniu*, tłum. T. Zajbt, Warszawa 2002.
2. Korol J., *Visual Basic w Excelu 2000: programowanie dla każdego*, Warszawa 2001.
3. Kuciński K., *ABC Excela*, Kraków 2001.
4. Langer M., *Excel 2002/XP PL*, tłum. G. Kowalczyk, Gliwice 2002.
5. Lasek M., Pęczkowski M., Boguszewski W., Ignatowska B., Mosdorf R., Siemieniuk N. Sienkiewicz J., *Excel w finansach i zarządzaniu*, Białystok 2000.
6. Simon J., *Excel. Profesjonalna analiza i prezentacja danych*, tłum. Ł. Suma, Gliwice 2006.

Do rozdziału 8:

1. Kopertowska M., *Grafika menedżerska i prezentacja*, Warszawa 1999.
2. Kuciński K., *ABC PowerPointa*, Kraków 2001.
3. Negrino T., *PowerPoint. Tworzenie prezentacji. Projekty*, tłum. A. Sikora-Godlewska, Gliwice 2005.

Do rozdziału 9:

1. Forte S., Howe T., Wall K., Kimmel P., Mullen R., *Access 2002. Projektowanie baz danych. Księga eksperta*, tłum. P. Gonera, Gliwice 2002.
2. Banachowski L., Chądzyńska A., Matejewski K., *Relacyjne bazy danych Wykłady i ćwiczenia*, Warszawa 2004.
3. Beynon-Davies P., *Systemy baz danych*, tłum. L. Banachowski, M. Banachowski, Warszawa 1998, 2003
4. Elmasri R., Navathe S.B., *Wprowadzenie do systemów baz danych*, tłum. B. Garbacz, B. Moczulski, M. Szczepaniak, Gliwice 2005.
5. Forte S., *Access 2000. Księga eksperta*, tłum. zbiorowe, Gliwice 2001.
6. Garcia-Molina H., Ullman J.D., Widom J., *Systemy baz danych: pełny wykład*, tłum. M. Jurkiewicz, Warszawa 2006.
7. Ladanyi H., *SQL. Księga eksperta*, tłum. A. Jakubowska, T. Żmijewski, Gliwice 2000.
8. Prague C.N., Irwin M. R., Reardon J., *Access 2003 PL. Biblia*, tłum. J. Gierlicki, G. Werner, T. Miszkiewicz, R. Meryk, Gliwice 2004.
9. Ray D.S., Ray E.S., *Po prostu Access 2002/XP PL*, tłum. M. Szolc, Gliwice 2003.
10. Ullman J.D., Widom J., *Podstawowy wykład z systemów baz danych*, tłum. M. Jurkiewicz, Warszawa 2001

Do rozdziału 10:

1. Biłat A., *Ontologiczna interpretacja logiki*, Lublin 2004
2. Dobrowolski G., *Technologie agentowe w zdecentralizowanych systemach informacyjno decyzyjnych*, Kraków 2002.

3. Hawkins J., Blakeslee S., *Istota inteligencji: fascynująca opowieść o ludzkim mózgu i myślących maszynach*, tłum. T. Walczak, Gliwice 2006.
4. Kluska-Nawarecka S., Dobrowolski G., Marcejan R., Nawarecki E., *Od pasywnych do aktywnych źródeł danych i wiedzy*, Kraków 2002.
5. Knosala R. i in., *Zastosowania metod sztucznej inteligencji w inżynierii produkcji*, Warszawa 2002.
6. Korol T., Prusak B., *Upadłość przedsiębiorstw a wykorzystanie sztucznej inteligencji*, Warszawa 2005.
7. Kosiński R.A., *Sztuczne sieci neuronowe: dynamika nieliniowa i chaos*, Warszawa 2004.
8. Kwaśnicka H., *Sztuczna inteligencja i systemy ekspertowe. Rozwój, perspektywy*, Wrocław 2005.
9. Łachwa A., *Rozmyty świat zbiorów, liczb relacji, faktów, reguł i decyzji*, Warszawa 2001.
10. Russell J.S., Norvig P., *Artificial Intelligence: A Modern Approach*, Prentice Hall, New Jersey 2002.
11. Rutkowska D., Piliński M., Rutkowski L., *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*, Warszawa 1997.

DO ROZDZIAŁU 11:

1. Bruner R., *Java w komercyjnych usługach sieciowych. Księga eksperta*, tłum. A. Fiącek, C. Welsyng, Gliwice 2003.
2. Crowder D.A., Bailey A., *Tworzenie stron WWW. Biblia*, tłum. T. Jarzębowicz, wyd. 2, Gliwice 2005.
3. Danowski B., *Świat według Mozilli. Firefox*, Gliwice 2005.
4. Danowski B., *Świat według Mozilli. Thunderbird*, Gliwice 2005.
5. Goldman D., *JavaScript. Księga eksperta*, tłum. A. Żytka, Gliwice 2000.
6. Otegem van M., *XSLT dla każdego*, tłum. T. Żmijewski, Gliwice 2003.
7. Reinhardt R., Warren Lentz J., *Flash 4. Księga Eksperta*, Gliwice 2000.
8. Sokół M., *Zaawansowane tworzenie stron WWW. Ćwiczenia praktyczne*, Gliwice 2001.
9. Sokół M., *Internet. Przewodnik*, Gliwice 2004.
10. Welling L., Thomson L., *PHP i MySQL. Tworzenie stron WWW. Vademecum profesjonalisty*, tłum. J. Smycz, D. Kaczmarek, wyd. 3, Gliwice 2005.

DO ROZDZIAŁU 12:

1. Heermann D.W., *Podstawy symulacji komputerowych w fizyce*, tłum. A. Adamczuk, Warszawa 1997
2. Motyka M., *Symulacje komputerowe w fizyce*, Gliwice 2002.
3. MSC. Patran 2001, User's Guide for Unix.
4. MSC/NASTRAN, Version 68.2, Quick Reference Guide.
5. MSC/NASTRAN, Version 68.2, Release Notes.
6. Stasiewicz A., *C++ Builder. Symulacje komputerowe*, Gliwice 2003.
7. Wojtuszkiewicz K., Zachara Z., *P Spice. Symulacje wzmacniaczy dyskretnych*, Warszawa 2002.

