

Komputer i sieci komputerowe



Rozdział

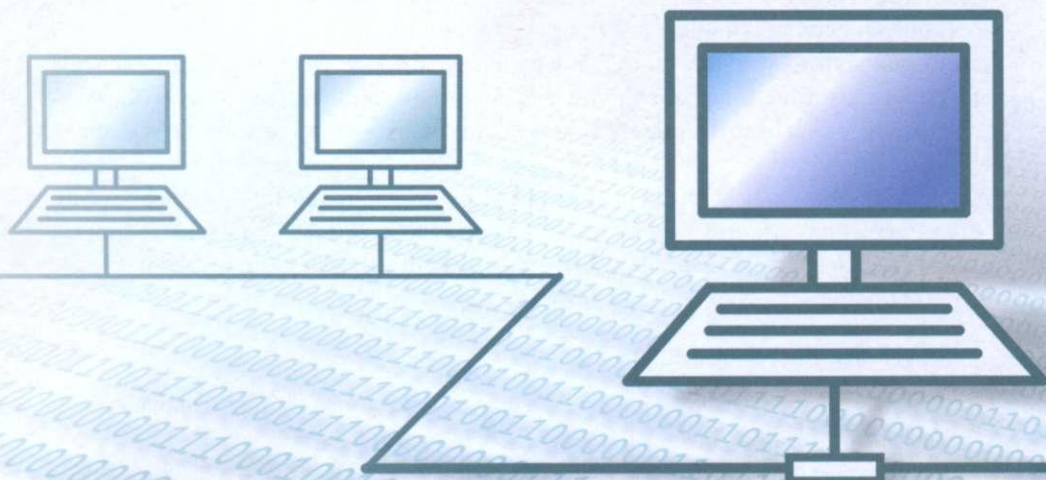
Temat 8. Reprezentacja danych w komputerze

Temat 9. Kompresja i szyfrowanie danych

Temat 10. Komputer i system operacyjny

Temat 11. Sieci komputerowe

Temat 12. Zadania projektowe



Reprezentacja danych w komputerze

1. Wprowadzenie do pozycyjnych systemów liczbowych

2. System dwójkowy (binarny)

- 2.1. Wartość liczby zapisanej w systemie dwójkowym
- 2.2. Wyznaczanie rozwinięcia dwójkowego liczby dziesiętnej

3. System szesnastkowy

- 3.1. Wartość liczby zapisanej w systemie szesnastkowym
- 3.2. Wyznaczanie rozwinięcia szesnastkowego liczby dziesiętnej
- 3.3. Zależność między systemem dwójkowym a szesnastkowym

4. Reprezentacja znaków



Warto powtórzyć

1. W jaki sposób w matematyce zapisuje się rozwinięcie dziesiętne liczby? Pokaż na przykładzie liczby 9871.
2. W jakim systemie liczbowym zapisywane są dane w komputerze? Omów, czym charakteryzuje się ten system.
3. Co to jest bit? Jakiej przyjmuje wartości?
4. Co to jest bajt? Z ilu składa się bitów?

1. Wprowadzenie do pozycyjnych systemów liczbowych

System liczbowy to sposób zapisywania i odczytywania liczb. Dwa podstawowe rodzaje systemów liczbowych to:

- **system addytywny** – zapisaną w nim **liczbę** odczytuje się jako sumę wartości jej poszczególnych **cyfr**; przykładem jest rzymski system liczbowy;
- **system pozycyjny** – o wartości cyfry decyduje jej miejsce w zapisie liczby; obecnie najczęściej stosowanymi pozycyjnymi systemami liczbowymi są: system dziesiętny, używany w życiu codziennym, oraz system dwójkowy, stosowany w systemach komputerowych.

Podstawą systemu dziesiętnego jest liczba 10, będąca jednostką drugiego rzędu. Jednostką trzeciego rzędu jest liczba 100, czwartego rzędu – liczba 1000 itd. Jednostką pierwszego rzędu jest liczba 1. Ta sama cyfra ma w systemie dziesiętnym różne znaczenie, w zależności od miejsca, w którym stoi w zapisie liczby.



Przykład 1. Pozycyjne systemy liczbowe

W liczbie **1476** zapisanej w systemie dziesiętnym cyfra „6” ma wartość 6, ponieważ znajduje się w kolumnie jednośc; cyfra „7” oznacza 70, ponieważ znajduje się w kolumnie dziesiątek; cyfra „4” oznacza 400, ponieważ znajduje się w kolumnie setek, a cyfra „1” oznacza 1000, ponieważ znajduje się w kolumnie tysięcy.

Czyli: $1476 = 1 \cdot 1000 + 4 \cdot 100 + 7 \cdot 10 + 6 \cdot 1$,

a przy użyciu zapisu potęgowego: $1476 = 1 \cdot 10^3 + 4 \cdot 10^2 + 7 \cdot 10^1 + 6 \cdot 10^0$.

Jeżeli ponumerujemy cyfry, zaczynając numerowanie od zera w kolejności od prawej strony, to zauważymy, że tak nadane numery cyfr odpowiadają kolejnym wykładnikom potęgi liczby 10 przy danej cyfrze:

3 2 1 0

$$1476 = 1 \cdot 10^3 + 4 \cdot 10^2 + 7 \cdot 10^1 + 6 \cdot 10^0$$

W systemie dziesiętnym używamy potęg liczby 10, podczas gdy np. w systemie trójkowym używamy potęg liczby 3, a w siedemnastkowym – potęg liczby 17. Potęgowaną liczbę nazywamy **podstawą systemu liczbowego**.

W systemie dziesiętnym do zapisywania liczb używamy dziesięciu cyfr. W systemie trójkowym są to trzy cyfry (0, 1, 2), a w systemie siedemnastkowym – siedemnaście (problem oznaczenia cyfr o wartościach większych niż 9 pokażemy dalej na przykładzie systemu szesnastkowego). Do zapisania liczby w systemie pozycyjnym o podstawie p konieczny jest zbiór cyfr $\{0, 1, \dots, p-1\}$.



W systemie pozycyjnym o podstawie p liczba n -cyfrowa: $(a_{n-1}a_{n-2}\dots a_1a_0)_p$ ma rozwinięcie:

$$(a_{n-1}a_{n-2}\dots a_1a_0)_p = a_{n-1} \cdot p^{n-1} + a_{n-2} \cdot p^{n-2} + \dots + a_1 \cdot p^1 + a_0 \cdot p^0, \quad [1]$$

gdzie: $a_{n-1}, a_{n-2}, \dots, a_1, a_0$ to kolejne cyfry danej liczby, p – podstawa systemu liczbowego.

2. System dwójkowy (binarny)

Sposób reprezentacji liczb stosowany w komputerach pokażemy na przykładzie liczb naturalnych. Prezentację liczb całkowitych ujemnych oraz liczb rzeczywistych omówimy w dalszej części podręcznika.

2.1. Wartość liczby zapisanej w systemie dwójkowym

Komputer posługuje się systemem binarnym, czyli dwójkowym. W systemie tym mamy do dyspozycji dwie cyfry: 0 i 1.

Jeśli przykładowa liczba ma postać: 11110, to taki zapis można zinterpretować jako jedenaście tysięcy sto dziesięć w systemie dziesiętnym lub jako liczbę zapisaną w systemie dwójkowym. Z tego powodu musimy uściślić zapis, przez określenie podstawy systemu liczbowego. Podstawę zapisujemy w indeksie dolnym na końcu liczby. Na przykład:

11110₂ – liczba w systemie dwójkowym,

20122₃ – liczba w systemie trójkowym,

11110, 20122 – liczby w systemie dziesiętnym.

Ponieważ na co dzień posługujemy się właśnie systemem dziesiętnym, przyjęło się, że liczby bez podstawy zapisanej na końcu to właśnie liczby dziesiętne. Czyli np. zamiast pisać 1476₁₀, piszemy po prostu 1476.



Aby obliczyć wartość dziesiętną liczby zapisanej w systemie dwójkowym, należy dokonać rozwinięcia liczby według wzoru [1], przyjmując jako podstawę systemu liczbę 2.



Przykład 2. Obliczanie wartości dziesiętnej liczby zapisanej w systemie dwójkowym

Obliczmy wartość liczby 11110_2 :

4 3 2 1 0

$$11110_2 = 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 =$$

$$= 1 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = 16 + 8 + 4 + 2 = 30.$$

Ponieważ $0 \cdot 2^n = 0$, oraz $1 \cdot 2^n = 2^n$, w praktyce wystarczy zsumować tylko te potęgi dwójki, które znajdują się przy cyfrach 1, np.:

11 10 9 8 7 6 5 4 3 2 1 0

$$101101110110_2 = 1 \cdot 2^{11} + 0 \cdot 2^{10} + 1 \cdot 2^9 + 1 \cdot 2^8 + 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 =$$

$$= 2048 + 512 + 256 + 64 + 32 + 16 + 4 + 2 = 2934.$$

W systemach komputerowych dane zapisywane są przy użyciu liczby bitów będącej kolejną potęgą liczby 2. Do zapisu liczb używa się: 8 bitów – tzw. krótkie słowo (1 bajt), 16 bitów – tzw. słowo, 32 bitów – tzw. długie słowo (podwójne słowo), 64 bitów – tzw. poczwórne słowo. Liczba 2934 byłaby więc zapisana jako:

MSB

LSB

0000101101110110₂ (dopełniona z lewej strony zerami).

W zapisie binarnym skrajna lewa wartość to tzw. najbardziej znaczący bit (z ang. MSB – *Most Significant Bit*), skrajna prawa – to tzw. najmniej znaczący bit (z ang. LSB – *Least Significant Bit*).



Ćwiczenie 1.

Oblicz wartość dziesiętną następujących liczb: 1000000000_2 , 11111111_2 .

2.2. Wyznaczanie rozwinięcia dwójkowego liczby dziesiętnej



Przykład 3. Obliczanie rozwinięcia dwójkowego liczby zapisanej w systemie dziesiętnym

Obliczmy rozwinięcie dwójkowe liczby 1476.

Liczba	Reszta	Komentarz
1476	0	$1476 = 2 \cdot 738 + 0$
738	0	$738 = 2 \cdot 369 + 0$
369	1	$369 = 2 \cdot 184 + 1$
184	0	$184 = 2 \cdot 92 + 0$
92	0	$92 = 2 \cdot 46 + 0$
46	0	$46 = 2 \cdot 23 + 0$
23	1	$23 = 2 \cdot 11 + 1$
11	1	$11 = 2 \cdot 5 + 1$
5	1	$5 = 2 \cdot 2 + 1$
2	0	$2 = 2 \cdot 1 + 0$
1	1	$1 = 2 \cdot 0 + 1$ iloraz równy 0 – koniec

Tabela 1. Obliczanie rozwinięcia dwójkowego liczby 1476

A zatem: $1476_{10} = 10111000100_2$



Aby wyznaczyć postać dwójkową (zwaną rozwinięciem dwójkowym lub binarnym) liczby dziesiętnej, powtarzamy operację dzielenia z resztą tej liczby przez dwa i zapisujemy wszystkie reszty z dzielenia, aż do otrzymania ilorazu równego 0.

Rozwinięcie dwójkowe liczby to zapisane kolejno reszty z dzielenia (począwszy od ostatniej).



Ćwiczenie 2.

Oblicz rozwinięcie dwójkowe liczb: 255, 1024.



Ćwiczenie 3.

Jaki otrzymasz rysunek, dokonując rozwinięcia dwójkowego (8-bitowego) następujących liczb i zapisując je jedna pod drugą: 16, 16, 56, 124, 254, 16, 16, 56?

3. System szesnastkowy

3.1. Wartość liczby zapisanej w systemie szesnastkowym

W praktyce posługiwanie się systemem dwójkowym jest bardzo niewygodne – zapis w nim jest długi i łatwo o pomyłkę (np. o pominięcie zera lub jedynek). Często więc w informatyce, zamiast systemu dwójkowego, wykorzystuje się system szesnastkowy, w którym do dyspozycji mamy 16 cyfr. Oprócz cyfr od 0 do 9 pojawiają się następujące cyfry:

A – o wartości 10, B – 11, C – 12, D – 13, E – 14, F – 15.

Przykłady liczb w systemie szesnastkowym: 100_{16} , AB_{16} , $FFFF_{16}$, $400D_{16}$.



Przykład 4. Obliczanie wartości liczby zapisanej w systemie szesnastkowym

Aby obliczyć wartość liczby zapisanej w systemie szesnastkowym, używamy wzoru [1], przyjmując za podstawę systemu liczbę 16:

2 1 0

$$100 = 1 \cdot 16^2 + 0 \cdot 16^1 + 0 \cdot 16^0 = 1 \cdot 256 + 0 \cdot 16 + 0 \cdot 1 = 256 + 0 + 0 = 256_{10}$$

3 2 1 0

$$ABCD_{16} = A \cdot 16^3 + B \cdot 16^2 + C \cdot 16^1 + D \cdot 16^0 = 10 \cdot 16^3 + 11 \cdot 16^2 + 12 \cdot 16^1 + 13 \cdot 16^0 = \\ = 10 \cdot 4096 + 11 \cdot 256 + 12 \cdot 16 + 13 \cdot 1 = 40960 + 2816 + 192 + 13 = 43981_{10}$$



Ćwiczenie 4.

Oblicz wartość dziesiętną następujących liczb szesnastkowych: $FFFF_{16}$, $C000_{16}$.

3.2. Wyznaczanie rozwinięcia szesnastkowego liczby dziesiętnej

Do zamiany z systemu szesnastkowego na system dziesiętny służy ten sam algorytm, z którego korzystaliśmy przy zamianie na system dwójkowy – z tą tylko różnicą, że dzielimy przez podstawę systemu, czyli liczbę 16, i zapisujemy reszty z dzielenia, aż do otrzymania ilorazu równego 0.



Przykład 5. Zamiana liczby zapisanej w systemie dziesiętnym na system szesnastkowy

Zapiszemy szesnastkowo liczbę 1476.

Liczba	Reszta	Komentarz
1476	4	$1476 = 16 \cdot 92 + 4$
92	12 (C)	$92 = 16 \cdot 5 + 12$
5	5	$5 = 16 \cdot 0 + 5$ iloraz równy 0 – koniec

Tabela 2. Zapisanie liczby 1476 w systemie szesnastkowym

Czytając od dołu, otrzymujemy: $1476_{10} = 5C4_{16}$.



Ćwiczenie 5.

Podaj swój wiek w systemie szesnastkowym.



Ćwiczenie 6.

Oblicz rozwinięcie szesnastkowe następujących liczb dziesiętnych: 100, 64.

3.3. Zależność między systemem dwójkowym a szesnastkowym

Systemu szesnastkowego można używać zamiast systemu dwójkowego, ponieważ $2^4 = 16$, a więc jedna cyfra szesnastkowa odpowiada **dokładnie** czterem cyfrom dwójkowym (w przypadku systemu dziesiętnego nie ma takiej bezpośredniej zależności pomiędzy cyframi dziesiętnymi a cyframi dwójkowymi, bowiem liczba 10 nie jest potęgą całkowitą liczby dwa). Aby zamienić liczbę szesnastkową na dwójkową, wystarczy więc sprawdzić wartość dziesiętną każdej cyfry szesnastkowej i zamienić ją na system dwójkowy, pamiętając o ewentualnym dopełnieniu zerami z przodu, tak by każda liczba dwójkowa odpowiadająca cyfrze szesnastkowej miała dokładnie cztery cyfry. Otrzymane rozwinięcia dwójkowe zapisujemy w tej samej kolejności, co odpowiadające im cyfry szesnastkowe.



Przykład 6. Zamienianie liczby zapisanej w systemie szesnastkowym na system dwójkowy

Zamienimy liczbę $5C4_{16}$ na system dwójkowy.

$$5_{16} = 5_{10} = 101_2 = 0101_2 \text{ (dopełniamy do 4 cyfr jednym zerem z przodu)}$$

$$C_{16} = 12_{10} = 1100_2$$

$$4_{16} = 4_{10} = 100_2 = 0100_2 \text{ (dopełniamy do 4 cyfr jednym zerem z przodu)}$$

Zatem:

$$5C4_{16} = \overset{5}{0101} \overset{C}{1100} \overset{4}{0100}_2$$

Aby przyspieszyć zamianę, można posłużyć się wartościami podanymi w tabeli 3.

Wartość dziesiętna	Wartość szesnastkowa	Wartość dwójkowa
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Tabela 3. Wartości szesnastkowe i dwójkowe liczb od 0 do 15

Zasadę, zgodnie z którą jedna cyfra szesnastkowa odpowiada czterem cyframi dwójkowym, wykorzystujemy także przy zamianie liczb z systemu dwójkowego na system szesnastkowy. Liczbę dwójkową dzielimy na czterocyfrowe grupy, zaczynając od prawej strony. Następnie obliczamy wartość dziesiętną każdej grupy i wpisujemy odpowiadającą jej cyfrę szesnastkową. Dla ułatwienia obliczeń można posłużyć się wartościami podanymi w tabeli 3.



Przykład 7. Zamienianie liczby zapisanej w systemie dwójkowym na system szesnastkowy

Zamienimy na system szesnastkowy liczbę 1100011101011101011011_2 .

$11100011110101111010111011_2$

11 0001 1101 0111 0101 1011 (dzielimy na grupy)

3 1 13 7 5 11 (obliczamy wartość dziesiętną każdej grupy)

3 1 D 7 5 B (wpisujemy odpowiednią cyfrę szesnastkową)

$1100011101011101011011_2 = 31D75B_{16}$



Ćwiczenie 7.

Zamień zapisy następujących liczb, używając systemu dwójkowego, dziesiętnego i szesnastkowego:

$$ABC_{16} = ?_{10} = ?_2$$

$$2331_{10} = ?_{16} = ?_2$$

$$1001111010101101110110_2 = ?_{16} = ?_{10}$$

4. Reprezentacja znaków

W postaci dwójkowej przedstawiane są nie tylko liczby, ale również inne znaki, np. litery, znaki interpunkcyjne, znaki kontrolne (nowa linia, spacja). Najbardziej rozpowszechnionym sposobem kodowania liter i innych znaków alfanumerycznych jest **kod ASCII** (z ang. *American Standard Code for Information Interchange*). Każda litera czy symbol posiada w tym kodzie swój odpowiednik liczbowy z zakresu <0, 255> (czyli zajmuje jeden bajt pamięci), np. spacja ma kod 32, znak „=” ma kod 61, znak „@” – kod 64, litera „a” – kod 97.

W programie **Tablica znaków** systemu Windows lub w oknie wstawiania symboli (opcja **Symbol** w menu **Wstaw** lub na karcie **Wstawianie**) edytora Microsoft Word można odnaleźć kod konkretnego znaku. W edytorze tekstu niektórym znakom przypisane są klawisze skrótu. Każdy znak można też wpisać, trzymając wciśnięty lewy klawisz **Alt** i wpisując na klawiaturze numerycznej kod ASCII (dziesiętny) tego znaku, poprzedzony zerem.



Ćwiczenie 8.

Korzystając z tabeli kodów ASCII, dostępnej w Internecie, programu Tablica znaków lub okna wstawiania symboli edytora tekstu, odszukaj kod znaku „§”. Wprowadź go, korzystając wyłącznie z klawiatury komputera.

Opracowany w Stanach Zjednoczonych kod ASCII nie uwzględniał w chwili swojego powstania znaków charakterystycznych dla języków innych niż angielski, np. polskich liter: ą, ę, ć, ó, ł, ń, ś, ż, ź. Próbowano kilkakrotnie rozwiązać ten problem przez zastąpienie rzadziej używanych w Polsce znaków amerykańskich polskimi znakami diakrytycznymi. Przez pewien czas w Polsce używano sześciu różnych sposobów kodowania. Ta różnorodność powodowała jednak sporo problemów z wymienialnością dokumentów, programów i z samym drukowaniem. Ostateczne rozwiązanie problemu znaków narodowych wprowadziła firma Microsoft w kolejnych wersjach systemu Windows, definiując strony kodowe dla poszczególnych krajów (polska strona kodowa ma numer 1250, mówimy więc o standardzie Windows-1250). Istnieje również międzynarodowy standard kodowania polskich liter: ISO-8859-2, niezgodny ze standardem Microsoftu.



Ćwiczenie 9.

Korzystając z Internetu, sprawdź kody ASCII polskich znaków w standardach Windows-1250 i ISO-8859-2.

Następcą kodu ASCII jest standard **Unicode**. W standardzie tym jeden znak zajmuje dwa bajty, co pozwala na zapisanie 65536 różnych znaków. Pierwszych 256 jest takich samych, jak w standardzie ASCII, pozostałe są przeznaczone dla alfabetów narodowych, w tym m.in. języka arabskiego, hebrajskiego, chińskiego i japońskiego.



Warto zapamiętać

- Wszelkie informacje (np. liczby, znaki alfanumeryczne) przechowywane w komputerze zapamiętywane są w postaci binarnej.
- Systemem pozycyjnym nazywamy taki sposób prezentowania liczb, w którym wartość cyfry zależy od jej pozycji w ciągu cyfr określającym liczbę.
- Rozwinięcie dwójkowe (inaczej binarne) liczby to zapisanie liczby dziesiętnej w systemie dwójkowym, czyli za pomocą zer i jedynek.



Pytania, problemy

1. Omów pozycyjne systemy liczbowe na przykładzie systemu dziesiętnego.
2. Jakie znasz niepozycyjne systemy liczbowe?
3. Dlaczego człowiek posługuje się systemem dziesiętnym?
4. Czy istnieje system jedynkowy? Jak wyglądałby zapis liczby w tym systemie?
5. W jaki sposób wyznaczamy rozwinięcie dwójkowe liczby dziesiętnej? Pokaż na przykładzie.
6. Jak oblicza się wartość dziesiętną liczby binarnej?
7. Dlaczego systemu szesnastkowego można używać zamiast systemu dwójkowego? Wyjaśnij na przykładzie (innym niż podany w temacie).



Zadania

1. Oblicz wartość dziesiętną następujących liczb: 1110011010100_2 , 3210_4 , 444_5 .
2. Oblicz rozwinięcie dwójkowe liczby określającej bieżący rok kalendarzowy.
3. Oblicz rozwinięcie dwójkowe liczby określającej twój wiek.
4. Oblicz rozwinięcie dwójkowe liczb: 2222, 500, 100.
5. Oblicz wartość dziesiętną następujących liczb szesnastkowych: ACE_{16} , 2000_{16} , 1111_{16} .
6. Oblicz rozwinięcie szesnastkowe następujących liczb dziesiętnych: 640, 32678, 1024.

Dla zainteresowanych

7. Napisz w wybranym języku programowania program obliczający wartość dziesiętną wprowadzonej z klawiatury liczby dwójkowej.
Uwaga: Zastanów się, w jaki sposób musi być zapamiętana liczba dwójkowa.
8. Zapisz w wybranym języku programowania iteracyjny algorytm zamiany liczby dziesiętnej na postać binarną.
9. Zapisz w wybranym języku programowania rekurencyjny algorytm zamiany liczby dziesiętnej na postać binarną.
10. Napisz w wybranym języku programowania funkcje **Hex2Dec** i **Dec2Hex**, umożliwiające zamianę liczb z systemu szesnastkowego na dziesiętny i odwrotnie. Zadbaj o właściwą postać parametrów i wyników funkcji.

Kompresja i szyfrowanie danych

1. Na czym polega kompresja danych?
2. Rodzaje kompresji
 - 2.1. Algorytmy kompresji stratnej
 - 2.2. Algorytmy kompresji bezstratnej
3. Szyfrowanie
 - 3.1. Szyfry podstawieniowe
 - 3.2. Szyfry przestawieniowe
 - 3.3. Szyfry z kluczem
 - 3.4. Szyfry z kluczem jawnym
 - 3.5. Jednokierunkowa funkcja skrótu
 - 3.6. Wykorzystanie algorytmów szyfrowania w podpisie elektronicznym



Warto powtórzyć

1. W jakim celu wykonuje się kompresję plików i folderów?
2. Jakich programów używamy do kompresji danych? Podaj przykłady.
3. Na czym polega szyfrowanie danych (temat C3, *Informatyka podstawowa*)?
4. Co to jest podpis elektroniczny (temat A5, *Informatyka podstawowa*)?

1. Na czym polega kompresja danych?

Kompresja (z łac. *compressio* – „ściśnięcie”) w różnych dziedzinach nauki oznacza zmniejszenie objętości (np. w fizyce – objętości cieczy, gazów). W informatyce kompresja odnosi się do zmniejszania objętości (wielkości) danych. Nie każde zmniejszenie objętości danych jest jednak kompresją.

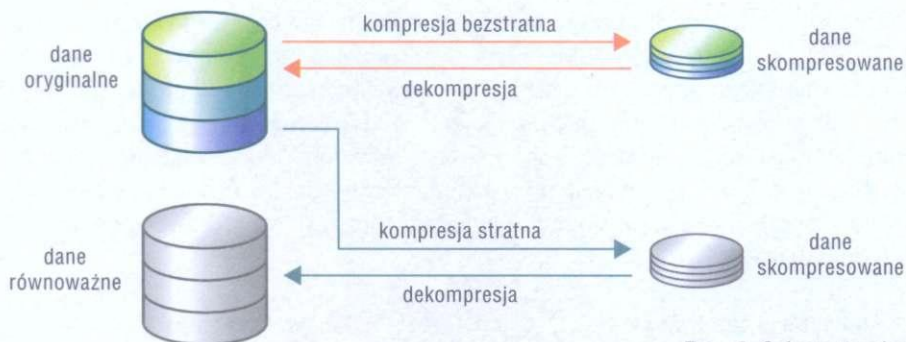


Kompresja to taki proces zmniejszania objętości danych, który umożliwia odtworzenie pierwotnych danych. Proces odtwarzania pierwotnych danych nazywamy **dekompresją**.

Dane odtworzone nie muszą być dokładnie takie same jak oryginalne – pewne metody kompresji powodują usunięcie niektórych fragmentów danych.

Dla każdego zbioru danych istnieje minimalna objętość, do jakiej dane te mogą zostać skompresowane. Oznacza to, że danych raz skompresowanych nie da się najczęściej ponownie skompresować.

Zastosowania kompresji danych są bardzo różnorodne. Bez kompresji danych nie istniałyby standardy JPEG, DVD, Blu-ray czy MP3. Kompresja pozwala także na efektywniejsze wykorzystywanie łącz telekomunikacyjnych (jest np. stosowana w modemach).



Rys. 1. Schemat wykonywania kompresji i dekompresji danych

Współczynnik kompresji obliczamy, dzieląc objętość danych skompresowanych przez objętość danych nieskompresowanych. Korzystamy z następujących wzorów:

$$R_c = \frac{V_k}{V_{nk}} \cdot 100\% \quad [1]$$

lub

$$R'_c = \left(1 - \frac{V_k}{V_{nk}}\right) \cdot 100\%, \quad [2]$$

gdzie:

R_c, R'_c – współczynniki kompresji,

V_k – objętość danych skompresowanych (w bajtach),

V_{nk} – objętość danych nieskompresowanych (w bajtach).

Wzór [1] pozwala nam obliczyć, jak bardzo dane zmniejszyły się w stosunku do oryginalnych, a wzór [2] – ile miejsca zaoszczędziliśmy w stosunku do oryginału.

Ze współczynnikiem kompresji wiąże się pojęcie **bitrate**. Stosuje się je w przypadku strumienia danych, a więc ciągu danych przesyłanych w czasie rzeczywistym (np. filmów, muzyki). W takim przypadku musimy dysponować urządzeniem zdolnym do przesłania w czasie jednej sekundy określonej liczby bitów. Przesłanie danych będzie możliwe tylko wtedy, gdy zostaną one odpowiednio zmniejszone. Bitrate podajemy najczęściej w bitach na sekundę, np. 128 Kb/s oznacza, że dane zostały zmniejszone tak, by jedna sekunda filmu lub dźwięku zajmowała nie więcej niż 128 kilobitów.

2. Rodzaje kompresji

Wyróżniamy dwa podstawowe rodzaje kompresji: **kompresję bezstratną** (ang. *lossless compression*) i **kompresję stratną** (ang. *lossy compression*).

W przypadku kompresji bezstratnej dane odtworzone są identyczne (bit po bicie) z danymi pierwotnymi. Natomiast w przypadku kompresji stratnej dane odtworzone są podobne do danych pierwotnych i na ogół różnią się od nich w sposób trudny do wychwycenia.

Z oczywistych względów kompresja stratna nie może być wykorzystywana do każdego rodzaju danych. Zmiana choćby jednej litery w przekazie tekstowym może prowadzić do całkowitej zmiany jego sensu. Podobnie zmiana nawet jednego bitu w programie komputerowym może spowodować zupełnie inne jego działanie (program może np. wykonać odejmowanie zamiast dodawania). Z drugiej strony, prawdopodobnie nikt nie zauważy niewielkiej zmiany koloru kilku pikseli obrazu komputerowego. Zastosowanie odpowiedniego rodzaju kompresji jest więc wymuszone przez rodzaj danych, które zamierzamy skompresować (tabela 1.).

Kompresja bezstratna	Kompresja stratna
• teksty • programy komputerowe • bazy danych • pliki z innymi danymi (pliki arkusza kalkulacyjnego, pliki konfiguracyjne, dane przesyłane w sieciach komputerowych itp.) • niektóre rodzaje grafiki (np. formaty GIF i TIFF)	• dźwięki (kompresja GSM wykorzystywana w telefonii komórkowej) • muzyka (format MP3) • obrazy (format JPEG) • filmy (format MPEG)

Tabela 1. Przykłady stosowania odpowiedniego rodzaju kompresji do rodzaju danych

2.1. Algorytmy kompresji stratnej

Algorytmy kompresji stratnej bazują na niedoskonałościach ludzkich zmysłów. Nie dostrzegamy np. niewielkich zmian barw lub drobnych różnic w fakturze powierzchni na obrazie. Jeżeli w danym momencie brzmi jednocześnie wiele dźwięków, słyszymy tylko niektóre z nich, itd. Algorytmy kompresji stratnej bazują na tym spostrzeżeniu i po prostu usuwają informacje o tych elementach, których nie jesteśmy w stanie dostrzec lub usłyszeć. Ponieważ takich informacji jest zaskakująco wiele, kompresja stratna jest bardzo efektywna.

Objętość pliku z obrazem lub dźwiękiem można zmniejszyć nawet dziesięciokrotnie bez zauważalnego dla obserwatora lub słuchacza pogorszenia jakości; dopiero przy dalszym zmniejszaniu objętości różnice zaczynają być odczuwalne. Do utraty jakości może też prowadzić wielokrotne powtarzanie cyklu kompresji i dekompresji, np. odczytywanie i zapisywanie obrazka w formacie JPEG.

2.2. Algorytmy kompresji bezstratnej

Algorytmy kompresji bezstratnej można podzielić na dwie grupy: **statystyczne** i **słownikowe**.

Algorytmy statystyczne operują na pojedynczych blokach danych (np. znakach tekstu lub fragmentach obrazu o określonej wielkości). Bazują one na takich faktach, jak np. ten, że znaki w tekście występują z różną częstością. W przypadku tekstu zapisanego w języku polskim najczęściej występują samogłoski „a”, „e”, „i”, „o”, litery takie, jak „ż” czy „ń” pojawiają się rzadko, natomiast znaki takie, jak „&” czy „#” występują zupełnie sporadycznie. Do zapisania każdego znaku przy użyciu kodu ASCII potrzebujemy dokładnie 8 bitów. Jeżeli jednak do zapisu najczęściej używanych znaków wykorzystamy ciągi bitów o mniejszej długości (np. 2 lub 3 bity), zaoszczędzimy dużo miejsca. Oczywiście,

aby każdy znak dało się jednoznacznie zidentyfikować, niektóre znaki trzeba będzie zapisać za pomocą więcej niż 8 bitów. Za pomocą tak długich ciągów można jednak zapisać znaki rzadko występujące, więc ostatecznie tego typu kompresja jest efektywna.

Przykładem kodowania tekstu opartego na częstości występowania znaków w tekście jest **alfabet Morse'a**. Znaki tego alfabetu są kodowane za pomocą dwóch symboli: kropki i kreski. Twórca alfabetu (Samuel Morse) przyjął zasadę, że kod danej litery jest krótszy, jeśli dana litera występuje w tekście częściej. Na przykład: kod litery B to: - • • •, a kod litery A, która występuje częściej niż B, to: • -. W alfabecie Morse'a nie są rozróżniane małe i wielkie litery.

Najbardziej znanym algorytmem kompresji statystycznej jest tzw. kodowanie Huffmana, w którym używa się wyłącznie zer i jedynek. Kody są tworzone tak, aby żaden nie był początkowym fragmentem innego (co pozwala na ich rozróżnianie) oraz aby średnia długość kodu danego znaku była możliwie najmniejsza.

Litera	Kod Morse'a	Litera	Kod Morse'a	Litera	Kod Morse'a	Litera	Kod Morse'a	Litera	Kod Morse'a
a	- •	f	• • • •	k	- • •	p	• • • •	u	• • •
b	- • • •	g	- • •	l	• • • •	q	- • • •	v	• • • •
c	- • • •	h	• • • •	m	- • •	r	• • •	w	• • •
d	- • •	i	• •	n	- •	s	• • •	x	- • • •
e	•	j	• • • •	o	- • •	t	-	y	- • • •
								z	- • • •

Tabela 2. Alfabet Morse'a – fragment z literami alfabetu łacińskiego



Przykład 1. Alfabet Morse'a

Aby odczytać tekst zakodowany za pomocą alfabetu Morse'a, wystarczy zastąpić kolejne kody odpowiednimi literami.

W alfabecie Morse'a zapis: • • / - • / • • - • / - - - / • - • / - • / - / - • - - • - / - • - / -
oznacza INFORMATYKA.

Uwaga: Znakiem „/” oddzielamy kody liter, a znakiem „-” – wyrazy.



Ćwiczenie 1.

Żeby dowiedzieć się, jaki tekst został zakodowany przez Samuela Morse'a w pierwszej depeszy nadanej w 1844 roku, odkoduj napis:

• - - / • • • • • / • - - / - / • • • • • / • - - / - / • • • • • / - - • / - - - / - • • / • - - / • - • / - - - / • • • - /
- - • / • • • • • / -



Ćwiczenie 2.

Zakoduj w alfabecie Morse'a swoje imię.

Algorytmy słownikowe operują na ciągach bitów o zmiennej długości (w szczególności na ciągach znaków). Patrząc np. na tekst zapisany w języku polskim, łatwo zauważyć, że pewne ciągi znaków często się powtarzają (np. w tym akapicie słowo „na” ze spacją pojawiło się już trzy razy). Teoretycznie można więc utworzyć słownik zawierający wszystkie możliwe ciągi znaków, a następnie zamiast znaków tekstu posługiwać się numerami

słów w słowniku. W praktyce tekst jest przeglądany od początku do końca i jeżeli algorytm natrafi na ciąg znaków, który pojawił się już wcześniej, zamiast całego tekstu zapisywana jest jedynie informacja o miejscu, w którym wcześniej określony ciąg znaków się pojawił.



Przykład 2. Stosowanie algorytmu słownikowego

Zdanie: *The rain in Spain falls mainly on the plain.* (44 bajty)

po skompresowaniu może mieć postać:

The rain <3,3>Sp<9,4>falls m<11,3>ly on t<34,3>p<15,3>. (33 bajty).

Zapis $\langle x, y \rangle$ (tzw. token) oznacza, że dana kombinacja znaków pojawiła się x znaków wcześniej i miała długość y znaków. W powyższym przykładzie do zapisania x wystarczy 6 bitów, natomiast do zapisania y potrzebne są 2 bity (maksymalne y to 4 bity, jednak nie ma sensu zaznaczać wystąpienia ciągu o długości 0 znaków).

Do zapisania jednego tokenu wystarczy więc 1 bajt.

Współczynnik kompresji dla powyższego przykładu wynosi $33/44 \cdot 100\% = 75\%$.

Najbardziej znanym algorytmem kompresji słownikowej, na którym bazują algorytmy wykorzystywane w programach takich, jak WinZIP, WinRAR, jest algorytm LZW (od nazwisk autorów: Lempel-Ziv-Welch).



Ćwiczenie 3.

Za pomocą metody słownikowej dokonaj kompresji zdania: „Moją pasją są kompresje i ich komputerowe wersje”. Oblicz współczynnik kompresji.

3. Szyfrowanie

Szyfr

to algorytm szyfrowania i deszyfrowania.

Szyfrowanie to technika umożliwiająca bezpieczne przesyłanie wiadomości. **Nadawca wiadomości** chciałby mieć pewność, że nikt niepowołany nie pozna treści wiadomości, ani też jej nie zmodyfikuje.

Niezaszyfrowaną wiadomość nazywamy **tekstem jawnym**, natomiast zaszyfrowaną – **szyfrogramem**.



Szyfrowanie to proces przetwarzania wiadomości, pozwalający na ukrycie jej tekstu, natomiast **deszyfrowanie** to proces odtwarzania treści szyfrogramu.



Rys. 2. Schemat procesu szyfrowania i deszyfrowania

Nauka zajmująca się zabezpieczaniem wiadomości to **kryptografia**, natomiast **kryptoanaliza** to nauka o łamaniu tychże zabezpieczeń. Razem tworzą one **kryptologię** – osobną dziedzinę matematyki. Szyfrowanie i deszyfrowanie wykonuje się za pomocą odpowiednich funkcji matematycznych, często przy użyciu komputera.

Szyfrowanie jest obecnie powszechnie wykorzystywane. Między innymi istota handlu elektronicznego opiera się na zapewnieniu poufności przesyłanych danych. Podobnie przed dostępem niepowołanych osób chronimy inne informacje: korespondencję osobistą, dane firmowe, wyniki badań itp.

3.1. Szyfry podstawieniowe



Szyfr podstawieniowy to taki, w którym każdy znak tekstu jawnego jest zastępowany innym znakiem.

Najbardziej znanym przykładem tekstu podstawieniowego jest **szyfr Cezara**, którego autorstwo przypisuje się Juliuszowi Cesarowi. Szyfr ten polega na zastąpieniu każdej litery tekstu jawnego znakiem stojącym w alfabecie o trzy pozycje w prawo względem znaku źródłowego. I tak „A” zostaje zastąpione przez „D”, „B” przez „E”, ... „W” przez „Z”, „X” przez „A”, „Y” przez „B” i „Z” przez „C”:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

X Y Z A B C D E F G H I J K L M N O P Q R S T U V W



Ćwiczenie 4.

Korzystając z szyfru Cezara, zaszyfruj zdanie: „Szyfr Cezara to najbardziej znany szyfr podstawieniowy”.

3.2. Szyfry przestawieniowe



Szyfr przestawieniowy to taki, w którym w tekście zaszyfrowanym pojawiają się wszystkie znaki tekstu jawnego, ale w innej kolejności.

Szyfr przestawieniowy najprościej zrealizować, zapisując tekst jawny wierszami o ustalonej długości, a czytając go kolumnami.



Przykład 3. Stosowanie szyfru przestawieniowego

Tekst jawny: TOBEORNOTTOBETHATISTHEQUESTION

T O B E O R N O T T

O B E T H A T I S T

H E Q U E S T I O N

Czytając kolumnami, otrzymujemy:

TOHOBEBEQETUOHERASNTTOIITSOTTN

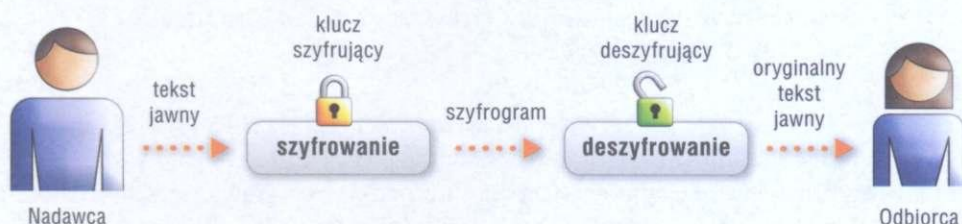
„To be or not to be –
that is the question”
– W. Szekspir, *Hamlet*,
akt III, scena 1.

3.3. Szyfry z kluczem

Wadą powyższych systemów szyfrowania jest to, że ich bezpieczeństwo jest uzależnione od tajności użytego algorytmu szyfrowania. Poznawszy algorytm, można odszyfrować dowolną zaszyfrowaną nim wiadomość. Wady tej są pozbawione **algorytmy z kluczem**.

W tego typu algorytmach do odczytania wiadomości potrzebna jest, oprócz znajomości sposobu szyfrowania, znajomość klucza. Znając sposób szyfrowania, lecz nie znając klucza, nie można odczytać wiadomości. Dodatkowo dla każdej wiadomości klucz może być inny, więc odczytanie jednej wiadomości nie oznacza, że będzie można odczytać również inne. Wszystkie nowoczesne sposoby szyfrowania wykorzystują klucze.

W zależności od algorytmu klucz szyfrujący może być taki sam jak klucz deszyfrujący lub też mogą one być różne.



Rys. 3. Schemat szyfrowania z kluczem

Przykładem prostego szyfru z kluczem jest **szyfr Vigenère'a**. W szyfrze tym każdy znak tekstu jawnego jest szyfrowany przy użyciu szyfru podstawieniowego, jednak dla każdego znaku przesunięcie w alfabecie jest inne i zależy od wartości odpowiedniego znaku klucza.



Przykład 4. Stosowanie szyfru Vigenère'a

Każdemu znakowi alfabetu (dla uproszczenia posłużymy się alfabetem łacińskim) przypiszemy określoną wartość liczbową: A – 1, B – 2, ... Z – 26.

Szyfrując słowo „TEKST” kluczem „KLUCZ”:

literę „T” zastąpimy literą leżącą w alfabecie o 11 (wartość „K”) miejsc dalej, literę „E” – literą leżącą o 12 (wartość „L”) miejsc dalej itd.

T ₂₀	E ₅	K ₁₁	S ₁₉	T ₂₀
K ₁₁	L ₁₂	U ₂₁	C ₃	Z ₂₆
E ₂₀₊₁₁₌₃₁₌₂₆₊₅	Q ₅₊₁₂₌₁₇	F ₁₁₊₂₁₌₃₂₌₂₆₊₆	V ₁₉₊₃₌₂₂	T ₂₀₊₂₆₌₄₆₌₂₆₊₂₀

Można posłużyć się tabelą 3.

		Znak tekstu jawnego																									
		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Znak klucza	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Tabela 3. Stosowanie szyfru Vigenère'a

Szyfrując w ten sposób, sumuje się wartości liczbowe każdego znaku tekstu jawnego i odpowiadającego mu znaku klucza. W ten sposób otrzymuje się wartość liczbową znaku szyfrogramu; jeżeli otrzymana wartość jest większa od 26, należy od niej

odjąć 26. Jeżeli długość klucza jest mniejsza od długości tekstu jawnego, klucz należy powielić odpowiednią ilość razy.

Deszyfrowanie polega na odjęciu od wartości liczbowej odpowiedniego znaku szyfrogramu wartości liczbowej odpowiedniego znaku klucza; jeżeli otrzymana wartość jest ujemna lub zerowa, należy dodać do niej 26. W ten sposób otrzymujemy wartość liczbową znaku tekstu oryginalnego.

Działający na podobnej zasadzie, choć oczywiście o wiele bardziej skomplikowany sposób szyfrowania, został wykorzystany przez Niemców przed II wojną światową w maszynie szyfrującej Enigma. Łamanie szyfrów Enigmy wymagało urządzeń obliczeniowych o dużej wydajności i było jednym z powodów budowy pierwszych komputerów. Szyfr Enigmy złamali w latach 30. XX w. polscy matematycy: Marian Rejewski, Jerzy Różycki i Henryk Zygalski.



Rys. 4. Niemiecka maszyna szyfrująca Enigma



Ćwiczenie 5.

Zaszyfruj nazwisko „KOWALSKI” kluczem „KLUCZ” za pomocą sposobu omówionego w przykładzie 4.

3.4. Szyfry z kluczem jawnym

Zasadniczą słabością algorytmów z kluczem jest to, że nadawca musi przekazać klucz odbiorcy. Wprawdzie mogą się oni wcześniej umówić, jakie klucze będą stosować, jednak i tak niezbędny jest kontakt nadawcy i odbiorcy. Z tego powodu proces komunikacji jest zawodny i nie zapewnia całkowitego bezpieczeństwa przekazu.

Wady tej pozbawione są algorytmy z **kluczem jawnym**. W algorytmach tych klucz szyfrujący jest inny niż klucz deszyfrujący. Klucz szyfrujący (zwany **kluczem jawnym** lub **kluczem publicznym**) można udostępnić każdemu, w celu zaszyfrowania wiadomości przeznaczonych dla odbiorcy. Jednak pasującym do klucza jawnego kluczem deszyfrującym (tzw. **kluczem tajnym** lub **kluczem prywatnym**) dysponuje tylko odbiorca wiadomości. Dzięki temu tylko on może odczytać zaszyfrowaną wiadomość.

Najbardziej znanym algorytmem z kluczem jawnym jest algorytm **RSA** (od nazwisk twórców: Rivest-Shamir-Adelman). Algorytm ten jest powszechnie stosowany w wielu produktach, takich jak pakiet szyfrowania PGP (ang. *Pretty Good Privacy*) czy protokół bezpiecznych połączeń SSL, wykorzystywany przez serwery i przeglądarki internetowe.

3.5. Jednokierunkowa funkcja skrótu



Jednokierunkowa funkcja skrótu przekształca dowolnie długi ciąg danych w krótki ciąg danych określonej długości.

Funkcja ta powinna być tak dobrana, by zmiana nawet jednego bitu w ciągu źródłowym powodowała zmianę także w ciągu wynikowym. Dzięki istnieniu takiej funkcji łatwe jest

np. sprawdzenie, czy podczas transmisji dane nie uległy celowemu bądź przypadkowemu zniekształceniu. Wystarczy bowiem obliczyć wartość funkcji skrótu dla przesłanego dokumentu i porównać ją z opublikowaną (np. w Internecie) wartością dla dokumentu oryginalnego. Jeżeli wartości się różnią – dokument został zmieniony.

Funkcje skrótu są też wykorzystywane w sieciowych systemach operacyjnych do zarządzania hasłami użytkownika. Hasło takie nie jest pamiętane nigdzie w pełnej postaci, a jedynie w postaci skróconej. Przy logowaniu hasło wpisane przez użytkownika jest skracane i porównywane z pamiętanym w systemie skrótem. Z postaci skróconej nie da się natomiast odzyskać oryginalnego hasła użytkownika, nie ma więc niebezpieczeństwa, że pozna go w ten sposób osoba nieupoważniona.

Najbardziej znane algorytmy tego typu to CRC, MD5 i SHA.

3.6. Wykorzystanie algorytmów szyfrowania w podpisie elektronicznym

Technologie szyfrowania umożliwiają także stosowanie **podpisu elektronicznego**.

Praktyczne formy podpisów elektronicznych stały się dostępne dzięki rozwojowi kryptografii z kluczem publicznym i polegają na dołączeniu do dokumentu jego skrótu (wygenerowanego za pomocą jednokierunkowej funkcji skrótu), zaszyfrowanego kluczem prywatnym strony podpisującej. Niezbędne jest zatem wygenerowanie pary kluczy: prywatnego i publicznego. Wygenerowany klucz publiczny przedstawia się następnie do **certyfikacji** właściwemu urzędowi. Po dostarczeniu wszystkich niezbędnych danych, potwierdzających naszą tożsamość, otrzymujemy certyfikat i możemy korzystać z podpisu elektronicznego.

Certyfikat umożliwia jednoznaczne związanie klucza publicznego z danymi posiadacza certyfikatu.

Klucz publiczny musi być udostępniony (jak nazwa wskazuje – upubliczniony). Takie klucze są przechowywane na specjalnych serwerach, tzw. serwerach kluczy.

Klucz prywatny przechowuje się zwykle na nośniku elektronicznym w postaci pliku, ale dodatkowo jest on zabezpieczony hasłem znanym tylko właścicielowi klucza. Klucz prywatny trzeba szczególnie chronić. O wygenerowanie klucza prywatnego i publicznego oraz otrzymanie certyfikatu można się zwrócić do urzędu certyfikacyjnego.

Wiadomość zaszyfrowana kluczem prywatnym może być deszyfrowana tylko właściwym (z tej samej pary) kluczem publicznym. Adresat tak deszyfrowanej wiadomości ma pewność, że pochodzi ona od nadawcy posiadającego klucz prywatny, zawarty w certyfikacie.

W sytuacji odwrotnej – wiadomość zaszyfrowana kluczem publicznym może być deszyfrowana tylko przez osobę posiadającą klucz prywatny. Taką wiadomość może przesyłać do nas każdy, kto zna nasz klucz publiczny. Nadawca ma pewność, że dotrze ona tylko do osoby posiadającej klucz prywatny (nie trafi w niepowołane ręce).



Warto zapamiętać

- Kompresję danych przeprowadzamy w celu minimalizacji objętości danych, np. przygotowując plik załącznika w poczcie elektronicznej.
- Wyróżniamy dwa rodzaje kompresji: bezstratną (odtworzone dane są identyczne z pierwotnymi) oraz stratną (odtworzone dane są podobne do pierwotnych).

- Aby zabezpieczać informacje, można posługiwać się różnymi algorytmami szyfrującymi, m.in. algorytmem podstawieniowym, przestawieniowym, szyfrem z kluczem.



Pytania, problemy

1. Podaj przykład kompresji statystycznej niezwiązanej z komputerami.
2. Na czym polega kompresja danych?
3. W jakim celu zwykle kompresujemy dane? Podaj kilka przykładów.
4. Wskaż na przykładzie różnice między kompresją stratną i bezstratną.
5. W jaki sposób działają techniki szyfrowania wiadomości? Podaj kilka przykładów.
6. Omów główną ideę algorytmów statystycznego i słownikowego.
7. Wyjaśnij na wymyślonym przez siebie przykładzie, na czym polega szyfrowanie podstawieniowe i przestawieniowe.
8. W jaki sposób wykorzystuje się szyfrowanie w podpisie elektronicznym?



Zadania

1. Odszyfruj tekst: „VCBIU FHCDUD MHVW SURVWB”.
2. Korzystając z metod szyfrowania omówionych w tym temacie, zaszyfruj na dwa sposoby swoje dane osobowe (imię, nazwisko, miejsce zamieszkania).
3. Przedstaw szyfrowanie Vigenère’a w arkuszu kalkulacyjnym.
Wskazówka: Każdą literę tekstu i klucza umieść w osobnej komórce, wykorzystaj funkcje KOD, ZNAK i JEŻELI.
4. Napisz w wybranym języku programowania program umożliwiający zaszyfrowanie bądź odszyfrowanie szyfrem Cezara tekstu wprowadzonego przez użytkownika.

Dla zainteresowanych

5. Napisz w wybranym języku programowania program obliczający i wypisujący częstotliwość występowania liter języka polskiego. Zmierz tę częstotliwość na podstawie tekstu *Pana Tadeusza* (znajdziesz go w Internecie).
6. Napisz w wybranym języku programowania program szyfrujący i deszyfrujący szyfrem przestawieniowym kolumnowym tekst wprowadzony przez użytkownika. Szerokość kolumny powinna być definiowana przez użytkownika.
7. **Steganografia** to nauka o ukrywaniu informacji wewnątrz innej informacji. Zapoznaj się z tym zagadnieniem, korzystając z dodatkowej literatury i źródeł internetowych.
8. Korzystając z Internetu i dodatkowej literatury, odszukaj informacje na temat Enigmy.
9. Korzystając z Internetu i dodatkowej literatury, dowiedz się więcej na temat:
 - a. kodu Huffmana,
 - b. algorytmu z kluczem jawnym RSA.

1. Logiczny model komputera
2. Na czym polega działanie procesora?
3. System operacyjny
4. Trochę historii – system MS DOS
5. Microsoft Windows
6. Mac OS
7. Unix
8. Linux
9. Systemy operacyjne dla urządzeń mobilnych



Warto powtórzyć

1. Wymień i krótko scharakteryzuj podstawowe elementy komputera.
2. Omów podział środków technologii informacyjnej ze względu na ich przeznaczenie.
3. Jaką rolę pełni w komputerze pamięć operacyjna?
4. Jaką funkcję pełni procesor? Jakie parametry wpływają na szybkość jego pracy? Czym charakteryzują się procesory wielordzeniowe?
5. Jak reprezentowane są dane liczbowe w komputerze? Pokaż na przykładzie.
6. Przedstaw klasyfikację oprogramowania.
7. Jaką rolę pełni system operacyjny? Przypomnij jego podstawowe funkcje.

Uwaga: W celu powtórzenia materiału przeczytaj temat C1, *Informatyka podstawowa*.

1. Logiczny model komputera

Od wprowadzenia komputerów osobistych na rynek minęło już ponad 30 lat, a od opracowania logicznego modelu komputera przez Johna von Neumanna – przeszło pół wieku. Ogólne zasady ich działania pozostały jednak niezmiennie i współczesne komputery są wciąż budowane według pierwotnej koncepcji.

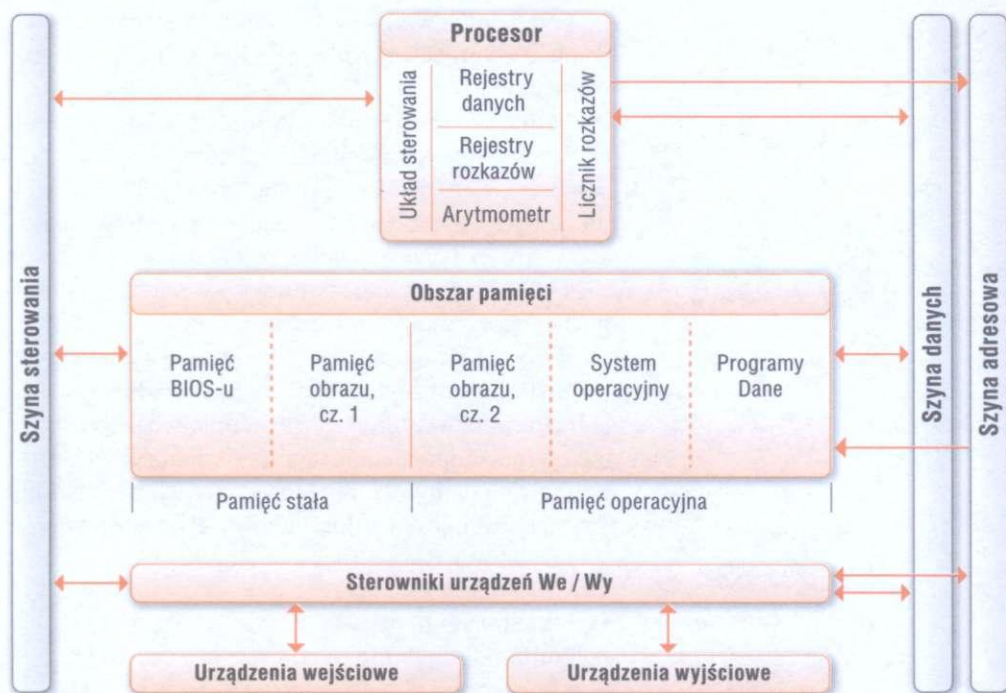


Idea maszyny von Neumanna:

- programy i dane posiadają taką samą postać,
- programy i dane są przechowywane w tej samej pamięci.

2. Na czym polega działanie procesora?

Głównym elementem komputera jest **procesor** (ang. CPU – *Central Procesor Unit*). Współczesny procesor to pojedynczy układ scalony, oparty na strukturze krzemowej, zawierającej od kilkudziesięciu milionów do ponad miliarda tranzystorów. Procesor wykonuje operacje arytmetyczne i logiczne oraz odpowiada za prawidłową współpracę wszystkich modułów funkcjonalnych komputera. Szybkość taktowania współczesnego



Rys. 1. Uproszczony model komputera zgodny z ideą von Neumanna

procesora wynosi kilka gigaherców, zatem w jednej sekundzie taki procesor potrafi wykonać kilka miliardów operacji (w przypadku procesorów wielordzeniowych liczbę tę należy pomnożyć przez liczbę rdzeni procesora). W ich zakres wchodzi m.in. operacje arytmetyczne (np. dodawanie, odejmowanie, porównywanie) i logiczne (np. suma logiczna, iloczyn, negacja) oraz operacje przesyłania danych z i do pamięci operacyjnej. Zbiór tych operacji nazywany jest **listą rozkazów procesora**. W oparciu o możliwości listy rozkazów konstruowane są programy komputerowe.



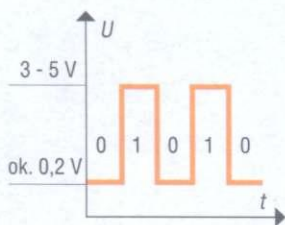
Rozkazy procesora to dane zapisane w postaci zero-jedynkowej, przy czym każdemu rozkazowi odpowiada inna sekwencja danych. Fizycznie rozkazy istnieją w komputerze w postaci impulsów elektrycznych.

Przyjmuje się, że obecność impulsu odpowiada liczbie 1, a brak impulsu – liczbie 0 (ze względu na właściwości materiałów, z jakich wykonane są elementy półprzewodnikowe, niskie napięcie jest zbliżone do zera). Każda z tych wartości (zarówno 0, jak i 1) stanowi 1 bit informacji. Kombinacje zer i jedynek pozwalają na matematyczne opisanie dowolnej informacji (temat 8.). Kombinacja ośmiu bitów to 1 bajt informacji.

Żeby komputer mógł wykonywać jakiegokolwiek programy i operować danymi, muszą być one dla niego dostępne. Miejsce, w którym są umieszczane, jest **pamięć operacyjna**. Choć fizycznie pamięć może być zrealizowana w różny sposób (jako pamięć

Funkcjonalne moduły procesora:

arytmometr, rejestr danych, rejestr rozkazów, licznik rozkazów, układ sterowania.



Rys. 2. Reprezentacja informacji w komputerze w postaci impulsów elektrycznych

Adres komórki	Zawartość komórki
0	
1	Rozkaz procesora
2	Rozkaz procesora
...	...
...	...
k-1	Wartość (dane)
k	Wartość (dane)
k+1	Adres innej komórki
...	...
...	Rozkaz procesora
n	...

Rys. 3. Dane umieszczone w komórkach pamięci o określonych adresach

półprzewodnikowa, dysk twardy lub pamięć taśmowa) – procesor widzi ją jako jednolity obszar pamięciowy.

Pamięć możemy sobie wyobrazić jako zbiór ponumerowanych komórek. Numer komórki jest jej adresem. Zawartość komórki stanowią dane. Danymi mogą być: wartości wprowadzone przez użytkownika, wartości będące wynikiem działania programu, rozkazy procesora, adresy innych komórek, wartości pochodzące z urządzeń wejścia.

Procesor komunikuje się z pamięcią przez **szyny** (magistrale): adresową i danych. Szyna adresowa jest jednokierunkowa. Za jej pośrednictwem procesor uaktywnia odpowiednie komórki pamięci, umożliwiając tym samym zapis lub odczyt ich zawartości. Szyna danych jest dwukierunkowa. Pozwala na przesyłanie danych do i z procesora.

W jaki sposób procesor wykonuje dodawanie liczb?

Wykonanie programu umieszczonego w pamięci operacyjnej zaczyna się od pobrania pierwszego kodu rozkazu tego programu. Procesor żąda dostępu do niego, wstawiając adres komórki, w której się znajduje, na szynę adresową. Pod żądanym adresem uaktywnia się odczyt zawartości komórki, zawartość ta jest przesyłana do procesora, gdzie zostaje umieszczona w rejestrze. Jeżeli jest to kod rozkazu, to zostanie on wykonany.



Przykład 1. Dodawanie liczb przez procesor

Dodawanie przez procesor wartości dwóch zmiennych (x , y) i zapisanie wartości w trzeciej zmiennej (z) dla: $x = 2$ i $y = 5$.

```
move.l (ax),d0 przesłanie liczby spod adresu ax (2) w pamięci do rejestru d0
move.l (ay),d1 przesłanie liczby spod adresu ay (5) w pamięci do rejestru d1
add.l d0,d1 wykonanie operacji sumowania i zapisanie wyniku w rejestrze d1
move.l d1,(az) przesłanie liczby z rejestru d1 (7) do adresu az w pamięci
```



W pamięci operacyjnej programu i dane umieszczane są tylko na czas pracy komputera.

Dane, które muszą być pamiętane, gdy komputer nie pracuje, zapisywane są w **pamięci masowej** (np. dysk twardy, płyta DVD).

3. System operacyjny

Na początku ery komputerów programy użytkowe były tworzone równocześnie z budową komputera – komputery budowano do wykonania konkretnego zadania, tworząc jednocześnie program, który miał je realizować. Oprogramowanie było zatem ściśle związane z budową komputera. Zmiana zastosowania komputera, by mógł wykonywać inne zadania, była niezwykle uciążliwa i kosztowna. Wymagała zmiany nie tylko programu, ale równocześnie przebudowy fragmentów maszyny. Rozwiązaniem tego problemu był pomysł zastosowania programu, który z jednej strony współpracowałby ściśle ze sprzętem, a z drugiej – reagował na działanie użytkownika. Zamiast podmieniać fragmenty maszyny – można podmieniać fragmenty programu pośredniczącego. Tak zrodził się pomysł **systemu operacyjnego**.

Pierwsze systemy operacyjne, choć nie były związane z konkretnym zadaniem, mogły obsługiwać tylko konkretne maszyny, a dokładnie serie (rodziny) maszyn. Producenci komputerów tworzyli systemy operacyjne na użytek produkowanych przez siebie maszyn. Musiało upłynąć jeszcze trochę czasu, zanim powstały tzw. **systemy otwarte**, związane jedynie z listą rozkazów procesora, a nie konstrukcją maszyny. Nie znaczy to, że systemy związane z konkretną maszyną straciły rację bytu. Istnieją i funkcjonują nadal, chociażby w generacji superkomputerów budowanych na potrzeby dużych centrów obliczeniowych. Są to potężnie rozbudowane programy, kontrolujące ogromną liczbę sprzętu centralnego i peryferyjnego, z którego korzysta centrum.

Uruchamianie takiego złożonego systemu – ładowanie systemu operacyjnego do superkomputera – trwa niekiedy kilka godzin. Na potrzeby komputerów osobistych opracowano systemy o wiele skromniejsze. Początkowo umieszczano je w pamięciach typu ROM, zwykle z jakimś prostym językiem programowania (dominował wówczas BASIC). Tak były budowane komputery ZX Spectrum, ATARI czy Commodore. Pojawienie się stacji dysków elastycznych oraz dysków twardych stało się impulsem do opracowania tzw. **dyskowych systemów operacyjnych**.

Oznaczenia:

B – bajt
b – bit
1 bajt = 8 bitów
1 Kb (kilobit) = 1024 b
1 Mb (megabit) = 1024 Kb
...
1 KB (kilobajt) = 1024 B
1 MB (megabajt) = 1024 KB
1 GB (gigabajt) = 1024 MB
1 TB (terabajt) = 1024 GB

System operacyjny

to zbiór programów kontrolujących pracę komputera oraz zainstalowanych w nim programów.



System operacyjny jest niezbędny, aby uruchomić jakikolwiek program w komputerze.

System operacyjny:

- pośredniczy pomiędzy użytkownikiem i komputerem oraz programami użytkowymi,
- zarządza pamięcią operacyjną i zasobami komputera,
- nadzoruje wymianę danych pomiędzy jednostką centralną i urządzeniami zewnętrznymi,
- organizuje zapis danych w pamięci masowej,
- sygnalizuje błędy w działaniu komputera i aplikacji,
- nadzoruje uruchamianie i wykonywanie programów.



Rys. 4. Ogólna struktura systemu operacyjnego na przykładzie Microsoft Windows

Główną część systemu operacyjnego stanowi **jądro**. Jest to zbiór najważniejszych podprogramów systemu. Jądro komunikuje się z urządzeniami zewnętrznymi za pośrednictwem sterowników, a z programami użytkownika – przez **powłokę systemu**.



Ćwiczenie 1.

Wymień funkcje systemu operacyjnego, z których najczęściej korzysta każdy użytkownik komputera. Na konkretnym przykładzie przedstaw ich zastosowanie.

W następnych punktach omówimy kilka systemów, które były kiedyś najbardziej popularne bądź są powszechnie stosowane w obecnie używanych komputerach i urządzeniach mobilnych.

4. Trochę historii – system MS DOS

Najbardziej znanym, obecnie rzadko już używanym systemem operacyjnym, jest MS DOS, który powstał w 1981 roku w firmie Microsoft i rozwijany był do roku 1995. Stanowił nieodłączne wyposażenie setek milionów komputerów osobistych, opartych na konstrukcji IBM PC. Kolejne jego wersje były coraz bardziej rozbudowane i coraz bardziej funkcjonalne. MS DOS jest systemem jednozadaniowym i przeznaczonym dla jednego użytkownika. Ma ograniczone możliwości zarządzania pamięcią, wymaga ograniczenia modułów programów użytkowych do 64 KB, nie posiada prawie żadnych możliwości graficznych. Z czasem dołożono do niego graficzną nakładkę i tak powstał system znany pod nazwą Microsoft Windows.

5. Microsoft Windows

System operacyjny Microsoft Windows był odpowiedzią firmy Microsoft na rosnące zapotrzebowanie na systemy zorientowane graficznie, a wprowadzone wcześniej w komputerach Apple Macintosh. Choć pierwsze dwie wersje systemu Windows przeszły prawie niezauważone, to wersja Windows 3 (z roku 1990) wywołała duże zainteresowanie, a prawdziwym bestsellerem okazała się wersja Microsoft Windows 3.11 for Workgroups, oferująca możliwość pracy w sieci.

Swój sukces Microsoft Windows zawdzięcza standaryzacji wielu elementów użytkowania programów. Do tej pory każdy producent oprogramowania tworzył własne standardy czcionek, własne systemy menu i nawigacji, własne polecenia. Każdy nowy program wymagał wiele czasu na przyswojenie sobie wszelkich nowości. Microsoft Windows wprowadził standaryzację wielu składników programów, dzięki czemu przyswojenie sobie nowego programu nie wymaga nauki od podstaw.

Microsoft Windows wprowadził też nowe mechanizmy zarządzania pamięcią, dzięki czemu można wprowadzić do niej kilka programów i używać ich równocześnie. Stąd

mówimy, że jest to system **wielozadaniowy**. Jeżeli któryś z programów nie mieści się w obszarze pamięci RAM, system Windows przepisuje go na dysk twardy do obszaru tzw. pamięci wirtualnej. Program w dalszym ciągu pozostaje aktywny, chociaż odwołania do niego są wolniejsze (czas dostępu do dysku jest dużo dłuższy niż odwołania do pamięci RAM).

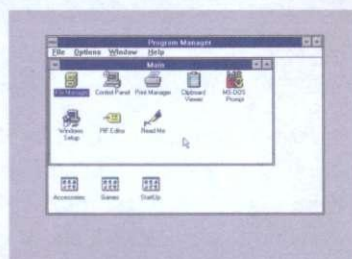
Microsoft Windows wprowadził ponadto takie nowości, jak DDE (ang. *Dynamic Data Exchange* – dynamiczna wymiana danych) i OLE (ang. *Object Linking and Embedding* – łączenie i osadzanie obiektów). Pozwalają one na współużytkowanie danych przez kilka aplikacji.

Kolejne wersje Microsoft Windows (95, 98 i Millennium) wprowadzały dalsze możliwości, ulepszenia i dodatki. Między innymi wprowadzono 32-bitowy system plików. Umożliwiono stosowanie długich (do 256 znaków) nazw plików i katalogów. Poszerzono możliwości współpracy z siecią. Dla zaawansowanych użytkowników rozwijano równoległe systemy profesjonalne: Microsoft Windows NT, Microsoft Windows 2000.

W roku 2002 do użytku weszła wersja systemu o nazwie Microsoft Windows XP, łącząca cechy systemów domowych (95, 98) i profesjonalnych (NT, 2000). Następcami systemu Windows XP były systemy: Windows Vista (2006 r.), Windows 7 (2009 r.) i Windows 8 (2012 r.). Nie wprowadzały one zasadniczych zmian w funkcjonowaniu systemu, natomiast modyfikacji ulegały elementy graficznego systemu komunikacji z użytkownikiem, poprawiono również zabezpieczenia systemu.

Wersja Windows 8 jest przystosowana do pracy z urządzeniami przenośnymi (smartfony, tablety). Interfejs użytkownika systemu Windows 8 jest nastawiony na pracę z ekranami dotykowymi – posiada interfejs kafelkowy, w którym menu **Start** zostało zastąpione kafelkami umieszczonymi na pulpicie (rys. 6.). Kafelki umożliwiają dostęp do aplikacji systemu Windows 8 poprzez ich kliknięcie lub dotknięcie. Aby jednak korzystać z opcji dotykowych, potrzebny jest tablet lub monitor z funkcją wielodotyku.

Użytkownik może łączyć kafelki w dowolne grupy. Dla użytkowników, którzy przyzwyczaili się do tradycyjnego pulpitu, istnieje możliwość korzystania także z niego.



Rys. 5. Zmiany graficznego interfejsu systemu Microsoft Windows (kolejno wersje: 1.0, 3.1, 98, XP, 7)



Rys. 6. Interfejs systemu Microsoft Windows 8 (ekran Start)



Rys. 7. Wyszukiwanie aplikacji w Windows 8

Nową funkcją wprowadzoną w Windows 8 jest pasek **Charms (Zakłęcia)** – na rysunku 6. widoczny w postaci kilku białych ikon po prawej stronie ekranu. Jest to menu, do którego dostęp jest możliwy w każdym momencie i z poziomu każdej aplikacji. Aby wyświetlić pasek **Zakłęcia**, wystarczy przesunąć kursor myszy lub palec (w przypadku ekranu dotykowego) od prawego górnego rogu ekranu w dół.

W Windows 8 usprawniono znajdowanie i otwieranie plików oraz wyszukiwanie aplikacji. Można wyszukiwać wszystkie elementy z poziomu ekranu **Start** (rys. 6.), wpisując je z klawiatury. Po wpisaniu pierwszego znaku, np. „p”, pokaże się ekran z wyświetlonymi aplikacjami, których nazwy zaczynają się na literę „p” lub ją zawierają (rys. 7.).

6. Mac OS

Zanim powstał Microsoft Windows, komputery Apple dysponowały już graficznym interfejsem użytkownika – GUI (ang. *Graphic User Interface*). Opracowany dla nich system operacyjny cechowała łatwość obsługi i instalacji. GUI był tak długo zmieniany, aż stał się przyjazny i czytelny nawet dla zupełnie początkujących użytkowników. Podjęcie produkcji komputerów tylko przez jedną firmę ograniczyło kłopoty związane z kompatybilnością. Instalacja i konfiguracja systemu nie sprawia większych problemów, dlatego komputery Mac uważane są za bardziej przyjazne dla mniej doświadczonych użytkowników.

Czy wiesz, że:

system Mac OS był podobno testowany przez babcię Steve'a Woźniaka (konstruktora komputera Apple)?



Rys. 8. Zmiany graficznego interfejsu systemu Mac OS (kolejno wersje: 1, 8.0, X 10.7)

7. Unix

Opracowany na początku lat 70. XX w. Unix jest jednym z najbardziej popularnych systemów operacyjnych. Pracuje zarówno na komputerach małych, jak i na superkomputerach. Istnieje kilka odmian tego systemu. Najbardziej znane to Solaris, SCO Unix, Irix, System V, Thru64 UNIX, UnixWare.

Unix jest systemem **wielodostępowym** i wielozadaniowym.

Posiada bardzo rozbudowane metody zabezpieczeń na każdym poziomie hierarchii katalogów i plików, potrafi obsługiwać wiele procesorów, jak również organizować systemy wielomaszynowe (**klastry**). Unix był pierwowzorem dla coraz bardziej popularnego dziś Linuksa.

System Unix miał znaczny wpływ na dzisiejszą informatykę, m.in. język C swoją popularność zawdzięcza właśnie temu systemowi.

8. Linux

Linux to system operacyjny, który zdobywa coraz większą popularność. Oparty jest na filozofii *open source*, tzn. system jest darmowy, a kod źródłowy systemu jest dostępny dla każdego. Linux dał się poznać jako system stabilny, dopracowany, o dużych możliwościach. Szczególnie chętnie jest wykorzystywany jako system dla serwerów internetowych. Również użytkownicy komputerów domowych decydują się na zainstalowanie Linuksa. Sprzyja temu dostępność coraz większej liczby aplikacji użytkowych, skutecznie zastępujących drogie programy platformy Windows. Specjalizowane wersje Linuksa wykorzystywane są w urządzeniach elektronicznych (np. routery, inteligentne telewizory itp.).

Linux dostępny jest w postaci tzw. **dystrybucji**. Stanowią one zbiór oprogramowania, zawierający system operacyjny (ściślej: jądro systemu) wraz z niezbędnymi sterownikami urządzeń oraz zestaw podstawowych programów użytkowych, takich jak programy biurowe, internetowe, proste gry i narzędzia systemowe. Najbardziej znane dystrybucje to Red Hat, SuSE, Debian i Ubuntu.

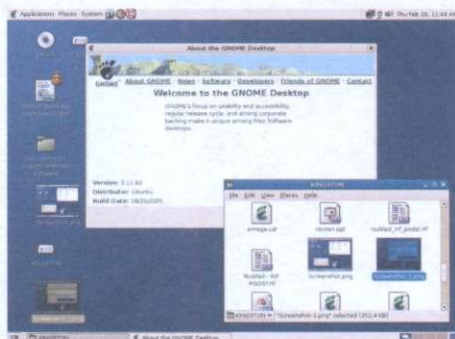
Podstawowe dystrybucje dostępne są bezpłatnie, można je np. pobrać z Internetu.

Linux może być sterowany za pomocą poleceń wprowadzanych przez **powłokę** (ang. *shell*) lub poprzez GUI. Najbardziej znane warianty GUI dostępne dla Linuksa to: KDE, Gnome, Xfce, Unity.

Największą popularność Linux zdobył jako system do obsługi serwerów. Jest powszechnie używany w serwisach internetowych do obsługi WWW i poczty elektronicznej. Coraz częściej spotyka się go w sieciach lokalnych w funkcji serwera plików i aplikacji, skutecznie wypierającego drogie serwery komercyjne Novella i Microsoftu.

Wielodostępowość

pozwała na korzystanie z tego samego komputera różnym użytkownikom, mającym swoje konta i własne ustawienia.



Rys. 9. Pulpit systemu Linux Gnome 2.11

9. Systemy operacyjne dla urządzeń mobilnych

Urządzeniami mobilnymi nazywamy urządzenia, za pomocą których można korzystać z zasobów Sieci, przetwarzać i przekazywać dane bez konieczności połączenia z Siecią za pomocą przewodów. Do urządzeń mobilnych zaliczamy np. telefony komórkowe, smartfony, tablety, palmtopy, notebooki, GPS. Rozwój tych urządzeń spowodował również rozwój systemów operacyjnych, które dla nich utworzono.

Android to system operacyjny dla urządzeń mobilnych, m.in. telefonów komórkowych, smartfonów, netbooków, tabletów. System ten bazuje na jądrze Linuksa i oprogramowaniu na licencji GNU. Jego twórcy wywodzą się z firmy Android Inc., kupionej w 2005 roku przez Google. W roku 2007 powstało konsorcjum OHA (*Open Handset Alliance*), którego celem jest rozwój **otwartych standardów** dla telefonii mobilnej. OHA zaprezentowało platformę Android.

Programiści tworzą coraz to nowe aplikacje przeznaczone dla użytkowników urządzeń mobilnych z systemem Android. W drugim kwartale 2012 roku urządzenia z systemem Android stanowiły 64% światowego rynku urządzeń mobilnych.

iOS to system operacyjny Apple Inc. dla urządzeń mobilnych iPhone, iPod touch oraz iPad. Bazuje na systemie operacyjnym Mac OS X 10.5. System iOS jest, w porównaniu z Androidem, mniej konfigurowalny, ale łatwiejszy w obsłudze.



Rys. 10. Interfejs systemu Android pokazany na wyświetlaczu tableta



Rys. 11. Interfejs systemu iOS pokazany na wyświetlaczu telefonu komórkowego



Warto zapamiętać

- Procesor jest układem scalonym, wykonującym operacje arytmetyczne i logiczne zgodnie z wprowadzonym programem.
- Procesor odpowiada za prawidłową współpracę wszystkich modułów funkcjonalnych komputera, tj. pamięci, dysków, kontrolera grafiki i urządzeń zewnętrznych.
- W procesorze są wykonywane proste operacje arytmetyczne i logiczne na danych pobieranych za pośrednictwem szyny adresowej i szyny danych.
- Działanie procesora polega na wykonywaniu kolejnych instrukcji programu.
- Systemy operacyjne to programy, które nadzorują pracę komputera i innych programów.
- Systemy operacyjne dla urządzeń mobilnych to np. Android, iOS.
- Porównanie systemów Linux i Windows przedstawiono w tabeli 1.

System Linux a Windows		
Dział	Linux	MS Windows
Cena	Darmowy	Komercyjny
Szybkość	Linux jest bardzo szybkim systemem, nie zużywa zbyt dużo pamięci RAM, a więc uruchamianie programów trwa krótko przez cały czas użytkowania systemu.	Uruchamianie programów w systemie Windows odbywa się w miarę szybko, jednak im dłużej użytkujemy system, tym dłużej czekamy na uruchomienie programu. Wiąże się to z wypełnieniem pamięci RAM i powierzchni dyskowej przez niepotrzebne dane.
Stabilność	Wysoka. Rzadko się zawiesza. Istnieje możliwość wyłączenia zawieszonego programu bez zamknięcia systemu.	Częste zawieszenia. Konieczność restartu.
Elastyczność	Możliwości dostosowania Linuksa do własnych potrzeb są bardzo szerokie – różne motywy, środowiska graficzne mogą całkowicie odmienić system. Także zabezpieczenia kont są bardzo dokładne. Każdy użytkownik ma ograniczoną możliwość manipulowania systemem, ale przy tym posiada możliwość rozbudowanej personalizacji.	Choć oferuje ustawianie różnych profili, personalizacja ogranicza się do ikon i tapety pulpitu. Cechują go słabe zabezpieczenia; każdy użytkownik może dowolnie ingerować w zawartość kont innych użytkowników.
Konfiguracja	Mogą wystąpić problemy przy konfigurowaniu karty graficznej oraz instalacji programów.	Bardzo prosta instalacja aplikacji. Instalacja niektórych sterowników urządzeń jest bardzo pracochłonna i czasami skomplikowana.
Multimedia	Słuchanie MP3 i oglądanie filmów na DVD jest możliwe.	Jakość filmów jest bardzo dobra. Odtwarzanie muzyki i filmów jest bardzo proste.
Biuro	Programy biurowe są dostępne za darmo.	Wiele programów biurowych ma niestety wysokie ceny.

Tabela 1. Porównanie systemów Windows i Linux



Pytania, problemy

1. Jaka jest główna idea maszyny von Neumanna?
2. Jak reprezentowane są rozkazy procesora?
3. W jaki sposób dane są umieszczane w pamięci operacyjnej?
4. Wyjaśnij, dlaczego przedstawiony w przykładzie 1. opis dodawania liczb przez procesor możemy nazwać algorytmem.
5. Przedstaw na schematycznym rysunku opis dodawania dwóch liczb przez procesor.
6. Jakie funkcje pełni system operacyjny?
7. Wymień i krótko scharakteryzuj powszechnie używane systemy operacyjne.
8. Omów system operacyjny używany w pracowni szkolnej.
9. Jakie systemy operacyjne są używane w urządzeniach mobilnych?
Omów przykładowe.

Dla zainteresowanych

10. Oceń przyszłość poszczególnych systemów operacyjnych.
11. Korzystając z dodatkowych źródeł, dowiedz się więcej na temat systemów operacyjnych stosowanych w urządzeniach mobilnych.
12. Jeśli instalowałeś system Linux w domu, podziel się swoimi doświadczeniami i uzupełnij o swoje spostrzeżenia tabelę porównującą ten system z MS Windows.
13. Uzupełnij tabelę 1, tak by zawierała porównanie wszystkich systemów operacyjnych omówionych w temacie. Dobierz odpowiednie kryteria porównawcze.
14. Wyszukaj informacje na temat zalet i wad systemu operacyjnego Android.
15. Wyszukaj informacje na temat zalet i wad systemu operacyjnego iOS.

1. Terminologia sieciowa – usystematyzowanie pojęć
2. Warstwowy model sieci
3. Określanie ustawień sieciowych danego komputera i jego lokalizacji w sieci
 - 3.1. Adres sieciowy
 - 3.2. Podsieci IPv4
 - 3.3. Protokół DHCP
 - 3.4. Protokół DNS
 - 3.5. Protokół TCP
 - 3.6. Określanie ustawień sieciowych komputera
4. Zasady administrowania siecią komputerową w architekturze „klient-serwer”



Warto powtórzyć

1. Dlaczego komputery łączą się w sieć?
2. Jaki jest podział sieci ze względu na wielkość?
3. Czym się charakteryzują modele sieci: „klient-serwer” i „każdy z każdym”?
4. Wymień podstawowe elementy, w które powinien być wyposażony komputer pracujący w sieci.
5. Co to jest protokół sieciowy?
6. Omów schemat przykładowej sieci domowej i szkolnej.
7. Na czym polega udostępnianie i mapowanie zasobów?
8. Jakich poleceń systemu operacyjnego używa się do przeprowadzenia podstawowej diagnostyki sieci komputerowej opartej na protokole TCP/IP?

Uwaga: W celu powtórzenia materiału przeczytaj temat C2, *Informatyka podstawowa*.

1. Terminologia sieciowa – usystematyzowanie pojęć

Wybrane pojęcia i metody dotyczące sieci komputerowych	
Sieć komputerowa	łączy ze sobą komputery w celu wymiany danych
Zasoby sieciowe	wszelkiego rodzaju dane i urządzenia, do których można uzyskać dostęp w obrębie sieci
Klient	komputer, który korzysta z zasobów
Serwer	komputer, który udostępnia zasoby
Switch	urządzenie sieciowe umożliwiające przesyłanie danych pomiędzy różnymi urządzeniami końcowymi w tej samej sieci
Router	urządzenie sieciowe umożliwiające przesyłanie pakietów danych pomiędzy różnymi sieciami komputerowymi tego samego typu
Brama (ang. <i>gateway</i>)	urządzenie sieciowe umożliwiające przesyłanie pakietów danych pomiędzy sieciami komputerowymi różnego typu (opartymi o różne protokoły)

Host	komputer lub inne urządzenie komputerowe pracujące w sieci TCP/IP	
Podział sieci ze względu na wielkość	Lokalne	obejmują komputery połączone na małym obszarze, np. w szkole, biurze, firmie
	Miejskie	łączy sieci lokalne na terenie jednego miasta
	Rozległe	łączy sieci lokalne na większym obszarze, np. miasta, kraju
	Internet	łączy ze sobą prawie wszystkie sieci na świecie
Podział sieci ze względu na model funkcjonowania	„Każdy z każdym”	każdy komputer będący składnikiem tego rodzaju sieci ma takie same prawa – może być równocześnie serwerem i klientem
	„Klient-serwer”	co najmniej jeden komputer pełni rolę serwera, udostępniając swoje zasoby wielu klientom
Podział sieci ze względu na topologię	Topologia gwiazdy	komputery połączone są ze sobą poprzez jeden punkt centralny
	Topologia szyny	komputery dołączone są do wspólnego przewodu, szeregowo lub równolegle
	Układ pierścienia	podobny do topologii szyny, ale końce przewodu są ze sobą połączone, tworząc pierścień
	Topologie łączone	komputery połączone z wykorzystaniem wielu różnych topologii; wykorzystywane w dużych sieciach komputerowych; najczęściej stosowana jest topologia wielu gwiazd
Konfiguracja sieciowa	Karta sieciowa	urządzenie umożliwiające połączenie z innymi komputerami
	Protokół	zbiór reguł, którym podlega komunikacja pomiędzy komputerami
	Klient sieci	oprogramowanie pozwalające korzystać z zasobów sieci pracującej pod kontrolą konkretnego systemu operacyjnego
	Usługa	określenie funkcji realizowanej przez komputer pracujący w sieci
Adres IP	unikatowy numer przyporządkowany urządzeniom sieci komputerowych (pracującym zarówno w Internecie, jak i w sieciach lokalnych) funkcjonującym w oparciu o protokół IP	
Protokół TCP/IP	najczęściej stosowany protokół (zbiór protokołów); używany zarówno w sieciach rozległych (Internet), jak i w sieciach lokalnych	
Praca w sieci	Logowanie	proces wprowadzania identyfikatora i hasła użytkownika, który rozpoczyna pracę w sieci
	Udostępnianie	zezwoenie na korzystanie z zasobów lokalnego komputera przez innych użytkowników sieci
	Mapowanie	przyporządkowanie oznaczenia dysku komputera lokalnego wybranemu zasobowi, znajdującemu się na zdalnym komputerze

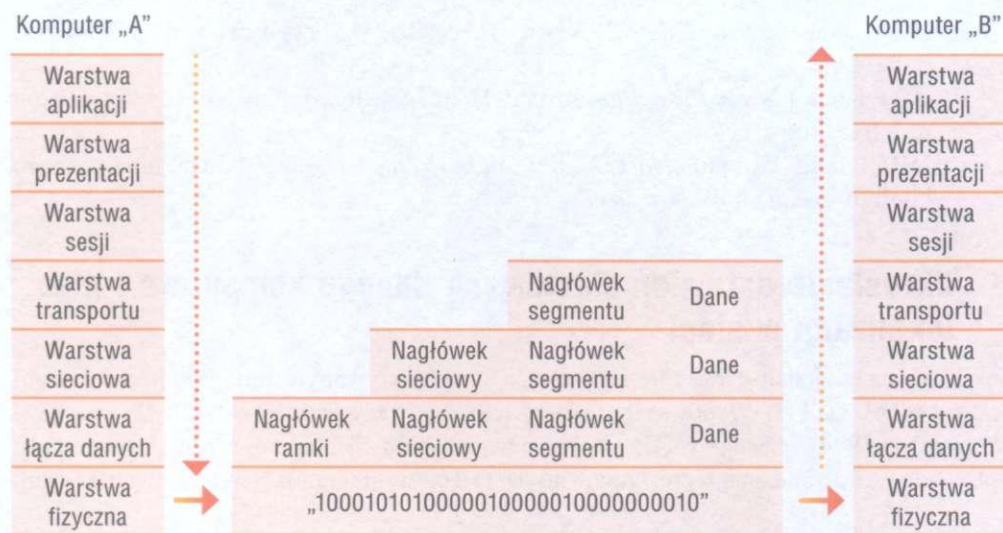
Tabela 1. Wybrane pojęcia i metody dotyczące sieci komputerowej

2. Warstwowy model sieci

Sieci komputerowe składają się z wielu elementów (zarówno sprzętowych, jak i programowych), pochodzących z różnych epok rozwoju informatyki, korzystających z różnych technologii i wyprodukowanych przez różnych producentów. Dla zapewnienia współpracy (w języku fachowym: interoperacyjności) wszystkich składowych sieci komputerowej stworzono teoretyczny model takiej współpracy, nazwany **modelem warstwowym OSI** (ang. *Open Systems Interconnection*).

W modelu OSI dane przesyłane pomiędzy systemami komputerowymi przechodzą przez siedem warstw (rys. 1.). Każda warstwa (z wyjątkiem warstwy sprzętowej) komunikuje się wyłącznie z warstwami znajdującymi się powyżej i poniżej za pomocą dobrze opisanych interfejsów. Te same warstwy dwóch systemów komputerów do komunikacji wykorzystują **protokoły komunikacyjne**.

Każda warstwa do danych otrzymanych z warstwy wyższej dodaje swoje własne, dokonuje zakodowania całości danych i ich podziału na mniejsze części (segmenty, pakiety, ramki), po czym przekazuje dane warstwie niższej. Warstwa najniższa, sprzętowa, odpowiada za fizyczne przesłanie danych do docelowego systemu komputerowego.



Rys. 1. Model warstwowy OSI

Warstwy przedstawiają sieć od strony funkcjonalnej. W każdej z nich definiuje się określone funkcje i usługi.

Warstwa aplikacji	Określa sposoby dostępu do środowiska sieciowego. Umożliwia udostępnianie plików, stron WWW, drukarek. Warstwa ta pełni rolę interfejsu między aplikacją użytkownika a usługami sieciowymi.
Warstwa prezentacji	Definiuje i standaryzuje format danych (np. znaki narodowe, liczby). Przeprowadza kodowanie i kompresję danych.
Warstwa sesji	Jest odpowiedzialna za prawidłową wymianę danych. Nadzoruje połączenie; wznowia je po przerwaniu.

Warstwa transportu	Zapewnia prawidłową kolejność pakietów po złożeniu ich w stacji odbiorczej. Sprawdza sumy kontrolne pakietów i wymusza powtórzenie transmisji po wykryciu błędu.
Warstwa sieciowa	Ustanawia połączenie między stacjami i dezaktywuje je po zakończeniu sesji. Jest odpowiedzialna za kierowanie ruchem pakietów w sieci (wyznacza trasy przebiegu pakietów).
Warstwa łącza danych	Zapewnia niezawodność łącza danych. Współpracuje ze sterownikami urządzeń sieciowych, np. kart sieciowych.
Warstwa fizyczna	Odpowiada za transmisję danych pomiędzy komputerami. Do warstwy tej należą urządzenia fizyczne, takie jak: karty sieciowe, modemy, koncentratory i media transmisyjne (kable, fale radiowe).

Tabela 2. Funkcje poszczególnych warstw sieci

Obecnie najczęściej wykorzystywane są sieci (w tym Internet) oparte o rodzinę protokołów znaną jako **TCP/IP**. Protokół **IP** (ang. *Internet Protocol*) realizuje warstwę sieciową, a protokoły **TCP** (ang. *Transport Control Protocol*) i **UDP** (ang. *User Datagram Protocol*) realizują warstwę transportu. Powszechnie wykorzystywane są też inne protokoły pomocnicze:

- **DNS** (ang. *Domain Name System*), wykorzystywany do zamiany nazw symbolicznych na adresy IP i odwrotnie,
- **ICMP** (ang. *Internet Control Message Protocol*), wykorzystywany do diagnostyki połączeń,
- **ARP** (ang. *Address Resolution Protocol*), służący do zamiany adresów logicznych na adresy fizyczne,
- **DHCP** (ang. *Dynamic Host Configuration Protocol*), służący do automatycznego konfigurowania hostów w sieci.

3. Określanie ustawień sieciowych danego komputera i jego lokalizacji w sieci

Jak już powiedzieliśmy, obecnie najczęściej wykorzystywanym protokołem sieciowym jest protokół TCP/IP. W powszechnym użyciu są dwie wersje protokołu IP: wersja 4 (oznaczana IPv4) i wersja 6 (IPv6). Podstawowa różnica pomiędzy tymi protokołami sprowadza się do sposobu reprezentacji **adresu sieciowego** urządzeń włączonych do sieci.

3.1. Adres sieciowy

W IPv4 adres sieciowy jest liczbą 32-bitową, co pozwala na zapisanie $2^{32} \approx 4$ miliardów adresów. Wobec zwiększania się ilości urządzeń pracujących w globalnej sieci komputerowej (Internecie) liczba ta jest niewystarczająca, by każdemu urządzeniu przypisać unikatowy adres (szacuje się, że dostępne adresy wyczerpią się w 2014 roku).

Specjalny adres 127.0.0.1 oznacza zawsze komputer lokalny, czyli ten, na którym wykonywane są operacje sieciowe.

W IPv6 adres sieciowy jest liczbą 128-bitową, co pozwala na zapisanie $2^{128} \approx 3,4 \cdot 10^{38}$ adresów. Niestety, IPv6 nie jest kompatybilny z IPv4 i nie jest obsługiwany przez większość urządzeń i systemów komputerowych włączonych do Internetu. W dalszej części tematu skoncentrujemy się więc na omówieniu najważniejszych cech protokołu IPv4.

Adres sieciowy IPv4 zapisujemy jako cztery liczby 8-bitowe (tzw. oktety), rozdzielone kropkami, np.:

77.79.216.19	77	79	216	19
	01001101	01001111	11011000	00010011

3.2. Podsieci IPv4

Ponieważ Internet nie posiada określonej struktury – jest raczej zbiorem wielu sieci, dzielących się na coraz mniejsze sieci – ważną rolę w określaniu lokalizacji komputera w sieci pełni **maska podsieci**. Maska podsieci określa, które bity adresu sieciowego oznaczają adres sieci, a które – urządzenia sieciowego. Bity maski ustawione na 1 określają adres sieci, a bity ustawione na 0 – adres urządzenia sieciowego. Po wykonaniu operacji logicznej AND na wszystkich odpowiadających sobie parach bitów adresu sieciowego i maski podsieci otrzymujemy adres podsieci.



Przykład 1. Wyznaczanie adresu podsieci na podstawie adresu IP i maski podsieci

	Dziesiętnie				Dwójkowo			
Adres sieciowy	192	168	112	100	11000000	10101000	01110000	01100100
Maska podsieci	255	255	255	0	11111111	11111111	11111111	00000000
Wynik operacji AND	192	168	112	0	11000000	10101000	01110000	00000000

Adres sieciowy 192.168.112.100 należy zatem do podsieci 192.168.112.0.

Przeznaczenie dwóch adresów każdej podsieci jest z góry określone: najmniejszy możliwy adres określa adres podsieci, największy możliwy – tzw. adres rozgłoszeniowy (ang. *broadcast*), tj. adres transmisji do wszystkich hostów w podsieci.



Przykład 2. Wyznaczanie adresów hostów w podsieci

Dla danych z przykładu 1:

adres sieci to 192.168.112.000

broadcast to 192.168.112.255

adresy hostów to 192.168.112.1-192.168.112.254.

Ponieważ w masce podsieci mogą wystąpić najpierw wyłącznie „1”, a potem wyłącznie „0”, maskę często opisuje się, podając liczbę „1” w masce.



Przykład 3. Skrócony sposób zapisu maski podsieci

Maska podsieci z przykładu 1. składa się z dwudziestu czterech cyfr 1.

Adres tej podsieci można więc zapisać jako 192.168.112.0/24.

Ważnym elementem każdej podsieci jest tzw. **brama domyślna** (ang. *default gateway*). Brama to wyróżnione urządzenie sieciowe, przez które przechodzi cała transmisja skierowana do urządzeń znajdujących się poza podsiecią. Adres bramy musi należeć do puli adresów hostów podsieci.



Przykład 4. Wyznaczanie adresu bramy dla podsieci

Dla sieci z przykładu 1. typowe adresy bramy to 192.168.112.1 albo 192.168.112.254.

Adresy IPv4 można podzielić na **publiczne** i **prywatne**. Adresy publiczne są dostępne (tzn. można do posługujących się nimi hostów przysyłać dane) dla wszystkich komputerów wchodzących w skład Internetu, natomiast adresy prywatne – tylko dla hostów wchodzących w skład danej podsieci.

W Internecie dostępne są trzy klasy adresów podsieci prywatnych:

Aby możliwy był dostęp z Internetu do usług udostępnianych przez komputer (komputery) w sieci lokalnej, w routerze niezbędne jest dokonanie tzw. mapowania portów, przyporządkowującego numery portów adresom IP z sieci lokalnej.

10.0.0.0/8	(2 ²⁴ adresów)
172.16.0.0/12	(2 ²⁰ adresów)
192.168.0.0/16	(2 ¹⁶ adresów).

Uwaga: Projektując sieć lokalną, powinniśmy zawsze używać adresów IP należących do podanych wyżej podsieci.

Aby możliwa była komunikacja pomiędzy hostami o adresach prywatnych a hostami o adresach publicznych (czyli Internetem), wykorzystuje się technikę **translacji adresów** (ang. *NAT – Network Address Translation*). Obsługuje ją większość routerów.

3.3. Protokół DHCP

Samodzielnie określanie adresów IP hostów przez użytkowników nie jest pożądane. Użytkownicy często nie mają odpowiedniej wiedzy, aby poprawnie ustalić adresy IP hosta i maskę podsieci, mogliby także przydzielać sobie adresy już wykorzystywane przez inne urządzenia sieciowe. Dlatego w większych sieciach komputerowych do nadawania adresów IP wykorzystuje się protokół DHCP. Dzięki jego zastosowaniu możliwe jest nie tylko nadanie hostom poprawnych adresów IP, ale także poprawne określenie adresów bramy domyślnej i serwera DNS, co jest niezbędne do zapewnienia dostępu do zasobów sieci.

Aby skorzystać z tego protokołu, niezbędne jest wybranie jednego lub kilku urządzeń sieciowych do pełnienia funkcji serwera DHCP (zwykle serwer DHCP jest elementem oprogramowania popularnych urządzeń sieciowych, np. routerów i punktów dostępowych).

3.4. Protokół DNS

Korzystanie z sieci (w tym Internetu) jest możliwe wyłącznie z wykorzystaniem adresów IP.



Ćwiczenie 1.

Wpisz w przeglądarce adres 74.125.26.103. Jaka strona wyświetliła się w przeglądarce?

Posługiwanie się wyłącznie adresami IP nie stwarza problemu komputerom, jest jednak kłopotliwe dla ludzi, wymaga bowiem zapamiętywania długich sekwencji numerów IP. W celu ułatwienia korzystania z sieci na początku lat 80. XX wieku wprowadzono protokół DNS, umożliwiający tworzenie mnemonicznych, łatwych do zapamiętania nazw hostów, zwanych **adresami domenowymi**. Nazwy te są automatycznie zamieniane na adresy IP (i odwrotnie – na podstawie adresu IP można zwykle ustalić adres domenowy).

Specjalny adres domenowy *localhost* odpowiada zawsze adresowi IP 127.0.0.1.

DNS charakteryzuje się zdecentralizowaną, rozproszoną i hierarchiczną strukturą. Odzwierciedleniem tej hierarchicznej struktury jest struktura adresu domenowego, w której poszczególne człony oddzielone są od siebie kropkami, przy czym element stojący najwyżej w hierarchii znajduje się po prawej stronie adresu. Każdy człon odpowiada innemu elementowi hierarchii.



Przykład 5. Struktura adresu domenowego

Adres *www.sejm.gov.pl* można odczytać:

strona *www* (*www*) – Sejmu RP (*sejm*) – który jest organizacja państwowa (*.gov*, z ang. *government*) – w Polsce (*pl*).

Po prawej stronie adresu znajduje się zawsze dwuliterowy kod kraju, w którym zarejestrowana jest domena albo jedna z domen najwyższego poziomu (z ang. *TLD – top-level Domain*). Do roku 2000 istniało siedem *TLD*:

- *com* – domeny komercyjne (np. firmy, przedsiębiorstwa),
- *org* – domeny organizacji niekomercyjnych (np. organizacje społeczne, partie polityczne, stowarzyszenia),
- *gov* – domeny jednostek administracji państwowej,
- *mil* – domeny wojskowe,
- *edu* – domeny jednostek edukacyjnych (np. szkoły, uniwersytety i inne instytucje oświatowe),
- *net* – domeny sieci komputerowych (np. dostawców Internetu),
- *int* – domeny organizacji międzynarodowych.

Po roku 2000 rozpoczęto proces dodawania nowych *TLD*. Obecne propozycje rozwoju zmierzają do możliwości tworzenia nowych *TLD* przez zainteresowane organizacje.

W Polsce nazewnictwem i rejestracją domen zajmuje się NASK (Naukowa i Akademicka Sieć Komputerowa).



Ćwiczenie 2.

Korzystając z Internetu, sprawdź, jakie *TLD* funkcjonują obecnie w sieci.



Ćwiczenie 3.

Jakim krajom odpowiadają następujące kody: *ca*, *cn*, *cz*, *fm*, *ru*, *tv*, *ua*, *uk*, *sh*?

Posiadanie własnej domeny wiąże się z koniecznością wniesienia stosownej opłaty, w wysokości od kilkudziesięciu do kilkuset złotych rocznie.

Pełne korzystanie z funkcji DNS możliwe jest w przypadku posiadania własnego serwera DNS. Darmowym programem tego rodzaju jest m.in. program BIND.

Aby ustalić adres IP hosta o podanym adresie domenowym, należy posłużyć się odpowiednim poleceniem systemu operacyjnego. W systemach Windows i Linux jest to polecenie **nslookup** *adres-domenowy*.

Większość systemów operacyjnych umożliwia definiowanie własnych adresów domenowych dla określonych adresów IP.

W systemach Windows należy w tym celu zmodyfikować zawartość pliku *C:\Windows\System32\drivers\etc\hosts*.

```

Wiersz polecenia
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Grazyna>nslookup www.migra.pl
Server: fritz.box
Address: 192.168.0.253

Non-authoritative answer:
Name: www.migra.pl
Address: 77.79.216.19

C:\Users\Grazyna>

```

Uwaga: Dla uzyskania właściwych efektów *adres-domenowy* należy zakończyć dodatkowo kropką.

3.5. Protokół TCP

Protokół TCP to protokół komunikacyjny, wykorzystywany do przesyłania danych pomiędzy procesami uruchomionymi na różnych hostach. Protokół TCP dba o poprawność przesyłania danych i ich integralność, dokonując m.in. retransmisji pakietów, które nie dotarły do hosta docelowego, i dbając o zachowanie właściwej kolejności pakietów docierających do hosta docelowego.

Jednym z ważniejszych pojęć związanych z protokołami warstwy transportu (czyli protokołów TCP i UDP) jest pojęcie **portu**. Port to liczba (z przedziału od 1 do 65535) używana do identyfikacji procesu odbierającego lub wysyłającego dane. Dzięki wprowadzeniu portów wiele procesów działających na jednym hoście może wysyłać dane (np. przeglądarka internetowa i klient poczty elektronicznej) lub odbierać dane (np. serwer WWW i serwer FTP). Często wykorzystywanym usługom przypisano standardowe numery portów (przykłady w tabeli 3).

Tabela 3. Standardowe numery portów przypisane wybranym usługom

Nr portu	Usługa
20	FTP – przesyłanie danych
21	FTP – przesyłanie poleceń
23	Telnet
25	SMTP
53	DNS
80	HTTP

Rys. 3. Wynik działania polecenia *netstat*. Parametr -n powoduje wyświetlanie adresów IP zamiast adresów domenowych oraz numerów portów zamiast ich nazw

```

Wiersz polecenia
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. Wszelkie prawa zastrzeżone.

C:\Users\Grazyna>netstat

Active Connections

Proto Local Address           Foreign Address         State
TCP    127.0.0.1:49291          HPK:49292              ESTABLISHED
TCP    127.0.0.1:49292          HPK:49291              ESTABLISHED
TCP    127.0.0.1:52320         HPK:wsd                TIME_WAIT
TCP    192.168.0.43:52161      OCSP:http              TIME_WAIT
TCP    192.168.0.43:52180      OCSP:http              TIME_WAIT
TCP    192.168.0.43:52293      SynologyNew:microsoft-ds ESTABLISHED
TCP    192.168.0.43:52300      OKI-C5000:http         TIME_WAIT
TCP    192.168.0.43:52301      HP2011:netbios-ssn     TIME_WAIT
TCP    192.168.0.43:52303      HP2011:netbios-ssn     TIME_WAIT
TCP    192.168.0.43:52304      fritz:49000            TIME_WAIT
TCP    192.168.0.43:52305      DLink:50000            TIME_WAIT
TCP    192.168.0.43:52307      fritz:49000            TIME_WAIT
TCP    192.168.0.43:52308      HP2011:wsd             TIME_WAIT
TCP    192.168.0.43:52310      OKI-C5000:http         TIME_WAIT
TCP    192.168.0.43:52311      DLink:50000            TIME_WAIT
TCP    192.168.0.43:52312      fritz:49000            TIME_WAIT
TCP    192.168.0.43:52313      MMS:wsd                TIME_WAIT
TCP    192.168.0.43:52314      fritz:49000            TIME_WAIT
TCP    192.168.0.43:52315      MMS:wsd                TIME_WAIT
TCP    192.168.0.43:52316      fritz:49000            TIME_WAIT
TCP    192.168.0.43:52317      HP2011:wsd             TIME_WAIT

```

Aby zobaczyć listę połączeń TCP aktywnych w danym komputerze, należy posłużyć się odpowiednim poleceniem systemu operacyjnego. W systemach Windows i Linux jest to polecenie **netstat**.

```
Wiersz polecenia
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. Wszelkie prawa zastrzeżone.
C:\Users\Grazyna>ipconfig

Windows IP Configuration

Ethernet adapter Połączenie sieciowe Bluetooth:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 
Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : fritz.box

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . : fritz.box
    Link-local IPv6 Address . . . . . : fe80::443b:cb34:205e:b07e%12
    IPv4 Address. . . . . : 192.168.0.43
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.253
```

Rys. 4. Wynik działania polecenia **ipconfig**

3.6. Określanie ustawień sieciowych komputera

Aby określić ustawienia sieciowe danego komputera, należy posłużyć się odpowiednim poleceniem systemu operacyjnego. W systemach Windows jest to polecenie **ipconfig**, a w systemach Linux – polecenie **ifconfig**.

Odpowiednie przełączniki pozwalają zmodyfikować efekty działania polecenia. Np. polecenie **ipconfig /all** wyświetli dodatkowo informacje na temat wszystkich kart sieciowych zainstalowanych w komputerze, w szczególności ich **adresy MAC** oraz informacje na temat serwerów DHCP i DNS.

Adres MAC

(ang. *Media Access Control*) to unikatowy w skali świata, 48-bitowy adres sprzętowy karty sieciowej komputera lub interfejsu sieciowego urządzenia sieci komputerowej.

```
Wiersz polecenia
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. Wszelkie prawa zastrzeżone.
C:\Users\Grazyna>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : HPGK
    Primary Dns Suffix . . . . . : 
    Node Type . . . . . : Hybrid
    IP Routing Enabled . . . . . : No
    WINS Proxy Enabled . . . . . : No
    DNS Suffix Search List . . . . . : fritz.box

Ethernet adapter Połączenie sieciowe Bluetooth:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 
    Description . . . . . : Urządzenie Bluetooth (sieć osobista)
    Physical Address. . . . . : 00-1A-6B-B8-55-8F
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . . : Yes

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : fritz.box
    Description . . . . . : Kontroler Realtek PCIe FE Family Controll
er
    Physical Address. . . . . : 00-1B-24-77-8D-E8
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . . : Yes
```

Rys. 5. Wynik działania polecenia **ipconfig** z użyciem opcji **/all**

4. Zasady administrowania siecią komputerową w architekturze „klient-serwer”

Podstawowym zadaniem administratora sieci komputerowej jest zapewnienie poprawnej i efektywnej pracy tej sieci. W szczególności można ten cel osiągnąć poprzez:

- zastosowanie odpowiednich elementów sprzętowych,
- podział sieci na mniejsze podsieci, odpowiadające strukturze fizycznej i logicznej danej organizacji (np. w każdym budynku jest inna sieć fizyczna, ale podziału logicznego dokonuje się ze względu na przeznaczenie komputerów działających w sieci),
- właściwą konfigurację komputerów sieciowych, w szczególności poprzez wykorzystanie protokołu DHCP,
- właściwe skonfigurowanie usług udostępnianych w sieci.

Proste sieci lokalne (np. domowe) tworzy się zazwyczaj z wykorzystaniem jednego urządzenia sieciowego, spełniającego funkcję:

- switcha (łączy urządzenia w warstwie fizycznej),
- punktu dostępowego (pozwala na dostęp do sieci urządzeniom bezprzewodowym),
- modemu (pozwala na fizyczne połączenie z dostawcą Internetu, np. poprzez łącze DSL),
- routera z funkcją NAT (pozwala komputerom z sieci lokalnej na dostęp do Internetu oraz udostępnia zasoby komputerów z sieci lokalnej w Internecie),
- firewalla (blokuje dostęp do określonych zasobów i usług),
- serwera DHCP,
- często także serwera plików (poprzez dołączenie dysku zewnętrznego).

The screenshot shows the web interface of a D-Link DIR-665 router. The browser address bar shows 'http://192.168.0.254/lan/setup'. The interface has a top navigation bar with tabs: SETUP, ADVANCED, STORAGE, TOOLS, STATUS, and SUPPORT. The 'SETUP' tab is selected, and the 'NETWORK SETTING' section is active. The left sidebar shows a tree view with 'Internet Setup', 'Network Setup', and 'LAN Setup'. The main content area contains several sections: 'NETWORK SETTING' with a note about DHCP and IP address changes, 'ROUTER SETTINGS' with fields for Router IP Address (192.168.0.254), Default Subnet Mask (255.255.255.0), Local Domain Name (inet.com.pl), and Enable DNS Relay (checked); 'DHCP SERVER SETTINGS' with 'Enable DHCP Server' checked and 'DHCP IP Address Range' set to 100 to 199; 'DHCP CLIENT LIST' with a table of connected devices; and '25 - DHCP RESERVATION' with a table for reserving IP addresses for specific computers.

Host Name	IP Address	MAC Address	Expired Time
Karma-PC	192.168.0.101	00:24:23:07:99:8E	Expired
HC-80400	192.168.0.11	00:1A:40:35:19:73	Never
HC-65500	192.168.0.12	00:24:06:A3:AF:78	Never
brnchvbr987ebbc241	192.168.0.31	78:84:3C:28:68:80	Never

Computer Name	IP Address	MAC Address
pc-4100	192.168.0.11	00:1A:40:35:19:73
pc-4100	192.168.0.12	00:24:06:A3:AF:78

Rys. 6.
Konfigurowanie
sieci lokalnej
i serwera DHCP
za pomocą
przeglądarki
internetowej

Konfiguracji tego typu urządzeń dokonuje się zwykle za pomocą przeglądarki internetowej, wpisując w przeglądarce nadany urządzeniu adres IP.

W większych sieciach lokalnych istnieje wiele urządzeń pełniących różne funkcje. W takich sieciach wydziela się zwykle jeden lub kilka komputerów do pełnienia funkcji serwera udostępniającego różne usługi. Oprócz wymienionych powyżej zastosowań mogą to być funkcje:

- serwera WWW, który udostępnia stronę internetową (tematy 31. i 32.),
- serwera FTP, który umożliwia przesyłanie plików,
- serwera poczty elektronicznej,
- serwera bazy danych (tematy 31. i 32.).

Serwery tego typu są zwykle specjalnymi komputerami o podwyższonych parametrach, działającymi pod kontrolą odpowiedniego systemu operacyjnego (systemy z rodziny Windows Server, np. Windows Server 2003, Windows Server 2008, lub specjalizowane dystrybucje Linuksa, np. Red Hat, Debian).

Zarządzanie takim serwerem wymaga zwykle biegłej znajomości systemu operacyjnego i zagadnień sieciowych, istnieją jednak liczne narzędzia ułatwiające to zadanie. Do takich narzędzi należy np. darmowy program Webmin, pozwalający zarządzać serwerem za pomocą przeglądarki internetowej.



Rys. 7. Konfiguracja serwera plików Samba za pomocą programu Webmin



Warto zapamiętać

- W modelu warstwowym sieci dane przesyłane pomiędzy systemami komputerowymi przechodzą przez siedem warstw: aplikacji, prezentacji, sesji, transportu, sieciową, łącza danych i fizyczną.
- Internet oparty jest na protokole TCP/IP. Powszechnie używa się dwóch wersji protokołu IP: wersji 4 (IPv4) i 6 (IPv6):
 - w IPv4 adres sieciowy jest liczbą 32-bitową,
 - w IPv6 adres sieciowy jest liczbą 128-bitową.
- Maska podsieci określa, które bity adresu sieciowego oznaczają adres sieci, a które – urządzenia sieciowego.
- W większych sieciach komputerowych do nadawania adresów IP wykorzystuje się protokół DHCP.
- Protokół DNS umożliwia tworzenie mnemonicznych, łatwych do zapamiętania nazw hostów, zwanych adresami domenowymi.
- Protokół TCP jest wykorzystywany do przesyłania danych pomiędzy procesami uruchomionymi na różnych hostach.
- Do określenia ustawień sieciowych w systemach Windows służy polecenie **ipconfig**, a w systemach Linux – polecenie **ifconfig**.



Pytania, problemy

1. Omów model warstwowy sieci, opisując krótko poszczególne warstwy.
2. Omów na przykładach z podręcznika dwie wersje protokołu IP (*IPv4* i *IPv6*).
3. Do czego wykorzystuje się protokół DHCP?
4. Czym są adresy domenowe? Omów na przykładzie.
5. Omów funkcje protokołu TCP.
6. W jaki sposób określa się ustawienia sieciowe?
7. Omów zasady administrowania siecią komputerową w architekturze „klient-serwer”.



Zadania

1. Jaki jest adres IP komputera, na którym pracujesz? Czy jest to adres prywatny, czy publiczny?
2. Jaki jest adres IP serwera obsługującego stronę internetową twojej szkoły?
3. Ile połączeń sieciowych jest aktywnych na komputerze, na którym pracujesz?

Dla zainteresowanych

4. Dokonaj odpowiedniego wpisu w pliku *hosts*, tak aby adres *www.wp.pl* wskazywał na stronę internetową *www.onet.pl*, a adres *www.onet.pl* – na stronę internetową *www.wp.pl*.

Wskazówka: Ustal adresy IP dla obu adresów domenowych, korzystając z polecenia **nslookup**.

1. Przygotowanie projektu
2. Wybrane normy etyczne i prawne
3. Rozwój informatyki i technologii informacyjno-komunikacyjnych
4. Szanse i zagrożenia związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych



Warto powtórzyć

1. Co jest przedmiotem prawa autorskiego?
2. Omów wybrane przepisy dotyczące korzystania z cudzych materiałów.
3. Jakie są zasady korzystania z programów komputerowych?
4. Podaj przykłady przestępczości komputerowej.
5. Jakie formy komunikacji i wymiany informacji oraz usługi rozwinęły się wraz z rozpowszechnieniem Internetu?
6. Jakie są zagrożenia wynikające z rozwoju technologii informacyjno-komunikacyjnych?
7. W jaki sposób możemy chronić się przed utratą danych i nieupoważnionym dostępem do komputera i sieci komputerowej?

Uwaga: W celu powtórzenia materiału i przygotowania projektów skorzystaj m.in. z tematów A1, A3, A4, A5, C1 i C3, *Informatyka podstawowa*.

1. Przygotowanie projektu

Zagadnienia dotyczące wybranych norm etycznych i prawnych, rozwoju informatyki i technologii informacyjno-komunikacyjnych (TIK) oraz związanych z tym rozwojem szans i zagrożeń, a także przestępczości komputerowej przedstawimy w formie projektów do samodzielnego wykonania (indywidualnego lub grupowego).

Etap pracy nad projektem

1. Określenie problemu.
2. Analiza problemu.
3. Ustalenie zadań szczegółowych.
4. Określenie szczegółowej formy poszczególnych dokumentów oraz dobór nazw plików.
5. Wybranie odpowiednich narzędzi – programów komputerowych.
6. Ustalenie założeń dotyczących łączenia dokumentów.
7. Przydzielenie zadań poszczególnym osobom.
8. Wykonanie zadań szczegółowych.
9. Połączenie dokumentów.
10. Ocena projektu – sprawdzenie, czy zadania zostały wykonane według wcześniejszych założeń.

W kolejnych punktach tematu przedstawiono zadania projektowe z krótkim opisem, przykładowymi zadaniami szczegółowymi i uwagami dotyczącymi ich wykonania.

W przypadku korzystania z utworów objętych prawem autorskim pamiętaj o przestrzeganiu przepisów prawa, dotyczących korzystania z cudzych materiałów.

Przed przystąpieniem do wykonania zadań szczegółowych należy zgromadzić odpowiednie materiały. Poza podręcznikiem *Informatyka podstawowa* można skorzystać z fachowej literatury, odpowiednich aktów prawnych (w przypadku przepisów prawa), Internetu, w tym z zasobów edukacyjnych udostępnianych na portalach edukacyjnych.

2. Wybrane normy etyczne i prawne

Projekt 1. Rozpowszechnianie programów komputerowych

Celem projektu jest przedstawienie podstawowych przepisów prawa, dotyczących rozpowszechniania programów komputerowych.

Analizę problemu należy rozpocząć od zapoznania się z przepisami prawa (odpowiednimi fragmentami). Przepisy prawa autorskiego reguluje ustawa z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (Dz. U. z 2006 r. Nr 90, poz. 631, z późn. zm.), dlatego tekst tej ustawy stanowić powinien podstawowe źródło informacji (teksty ustaw można znaleźć np. na stronie Internetowego Systemu Aktów Prawnych Sejmu RP: <http://isap.sejm.gov.pl/>).

Zaczynamy od wyjaśnienia, co jest przedmiotem prawa autorskiego i co nie podlega prawu autorskiemu, kiedy utwór określamy jako rozpowszechniony, a kiedy jako opublikowany. Omówienia wymagają przepisy dotyczące korzystania z cudzych materiałów, ze szczególnym uwzględnieniem utworów rozpowszechnionych (tu: programów komputerowych). Należy przedstawić też rodzaje licencji na programy komputerowe.

Zadania szczegółowe

1. Terminologia prawa autorskiego, m.in.: *prawo autorskie, przedmiot prawa autorskiego, autor, utwór, autorskie prawa osobiste, autorskie prawa majątkowe, utwór rozpowszechniony, utwór opublikowany, licencja*.
2. Korzystanie z cudzych utworów, w tym programów komputerowych. Wyjaśnienie, kiedy można skorzystać z cudzego utworu bez pytania o zgodę i jakie kary grożą za rozpowszechnianie cudzego utworu.
3. Zasady korzystania z programów komputerowych, w tym różne typy licencji na programy komputerowe.

Uwagi dotyczące realizacji projektu

- Dokument tworzony w zadaniu 1. można opracować w formie słowniczka terminów prawnych, umieszczonego na ostatnich stronach (lub slajdach) projektu. W miejscu pierwszego pojawienia się danego hasła w dokumencie wystarczy umieścić hiperłącze do jego definicji umieszczonej w tym samym dokumencie. Takie rozwiązanie jest możliwe w przypadku przygotowywania projektu w postaci prezentacji multimedialnej, jak i w formie dokumentu tekstowego, tworzonych w edytorze tekstu.
- Opracowując zadanie 2, należy wyjść od odpowiednich przepisów prawnych (wybranych artykułów ustawy o prawie autorskim i Kodeksu karnego). Przepisy te należy zilustrować przykładami i opatrzyć własnym komentarzem.
- Zagadnienie licencji na programy komputerowe trzeba omówić szczególnie dokładnie, podając nie tylko podstawowe rodzaje licencji, ale także wyjaśniając, na czym polega korzystanie z danej licencji, i posługując się konkretnymi przykładami.

- Proponowana liczba osób w grupie: 3-5; w przypadku pięciu osób: zadanie 1. opracowuje jedna osoba, a zadania 2. i 3. – po dwie osoby.

Projekt 2. Bezpieczeństwo i ochrona danych oraz informacji w komputerze i w sieciach komputerowych

Celem projektu jest zaprezentowanie metod zapewniania bezpieczeństwa i ochrony danych oraz informacji w komputerze i w sieciach komputerowych.

Analizę problemu należy rozpocząć od sklasyfikowania niebezpieczeństw, z jakimi możemy się spotkać, a następnie określenia sposobów ochrony przed nimi (np. przed złośliwym oprogramowaniem chronią nas programy zabezpieczające komputer).

W projekcie należy uwzględnić najnowsze osiągnięcia w zakresie ochrony danych przed pojawiającymi się – ciągle nowymi – zagrożeniami. Trzeba też wyjaśnić, w jaki sposób odzyskuje się utracone dane.

Zadania szczegółowe

1. Ochrona danych przed nieupoważnionym dostępem powstałym w wyniku nieświadomych działań użytkownika i innych osób.
2. Rodzaje złośliwego oprogramowania.
3. Sposoby zabezpieczenia komputera przed nieupoważnionym dostępem, w tym wskazanie odpowiedniego oprogramowania i sposobów szyfrowania danych.
4. Metody odzyskiwania utraconych danych.

Uwagi dotyczące realizacji projektu

- Opracowując zadanie 1, należy szczególną uwagę zwrócić na serwisy społecznościowe, w których często występuje sytuacja nieświadomego udostępniania danych innym osobom. Poza ogólnymi zasadami postępowania w celu ochrony naszej prywatności, warto podać też przykładowe skutki beztroskiego zachowania w Sieci.
- Opracowanie zadań szczegółowych 1-3 można połączyć, wypełniając tabelę:

Rodzaj zagrożenia	Sposoby ochrony, czyli jak się ustrzec lub jak zapobiegać
Phishing sposób zdobywania poufnych informacji oparty na inżynierii społecznej – poprzez podszywanie się.	Nie należy klikać na linki znajdujące się w otrzymanych wiadomościach; w razie potrzeby należy wejść na stronę, wpisując jej adres w przeglądarce lub wybierając dodaną przez siebie zakładkę. W przypadku bezpiecznych połączeń należy zawsze sprawdzać certyfikat strony i nie kontynuować procesu logowania, jeżeli przeglądarka informuje, że certyfikat jest niepoprawny.
...	...

- Treść tak przygotowanej tabeli można umieścić na slajdach prezentacji, np. na jednym slajdzie jeden wiersz tabeli, a pokaz przeprowadzić z podziałem na role: jedna osoba omawia rodzaj zagrożenia, a druga – sposób ochrony. Opisy umieszczane na slajdach należy odpowiednio skrócić.
- Zadanie 4. może stanowić oddzielną prezentację, podsumowującą cały projekt, ale można też dołączyć opracowanie zagadnienia do tabeli, tworząc jedną prezentację.
- Proponowana liczba osób w grupie: 4-6.

Projekt 3. Przestępczość komputerowa

Celem projektu jest omówienie zagadnienia przestępczości komputerowej, w tym piractwa komputerowego i nielegalnych transakcji w Sieci.

Analizę projektu należy rozpocząć od określenia, czym jest przestępstwo komputerowe w rozumieniu przepisów prawa. Następnie należy zapoznać się z rodzajami takich przestępstw i wyszukać najnowsze informacje na temat przestępczości komputerowej. Należy też spróbować odszukać, jakie kary grożą za wybrane przestępstwa. W tym celu można skorzystać z odpowiednich przepisów Kodeksu karnego lub informacji umieszczonych na fachowych stronach w Internecie.

Zadania szczegółowe

1. Przykłady przestępstw komputerowych, m.in.: piractwo komputerowe, hacking, podsłuch komputerowy, bezprawne niszczenie informacji, oszustwo komputerowe, sabotaż komputerowy, szpiegostwo komputerowe, podrzucanie wirusów i rozsyłanie innych szkodliwych programów.
2. Wybrane przepisy prawa, dotyczące kar za przestępstwa komputerowe, na przykład za piractwo komputerowe czy podrzucanie wirusów.

Uwagi dotyczące realizacji projektu

- Projekt można przygotować w postaci prezentacji multimedialnej i połączyć (w miarę możliwości) opis danego przestępstwa z przewidywaną karą za jego popełnienie.
- Jeśli są takie możliwości, warto odnieść się do kilku rzeczywistych przykładów łamania prawa w zakresie przestępczości komputerowej. Konkretnie przykłady bardziej przemawiają do słuchaczy niż same definicje.
- Proponowana liczba osób w grupie: 2-4.

3. Rozwój informatyki i technologii informacyjno-komunikacyjnych

Projekt 4. Najważniejsze elementy procesu rozwoju informatyki i technologii informacyjno-komunikacyjnych

Celem projektu jest przedstawienie najważniejszych elementów procesu rozwoju informatyki i technologii informacyjno-komunikacyjnych.

Punktem wyjścia powinna być krótka historia rozwoju informatyki – chronologiczne ujęcie ważniejszych wydarzeń. Można zacząć od wynalezienia abakusa (ok. X w. p.n.e.) i wspomnieć o pierwszych kalkulatorach mechanicznych (XVII w.), a następnie elektrycznych. Warto odpowiedzieć na pytanie, co tamte maszyny miały wspólnego z dzisiejszymi komputerami, wykonującymi nie tylko bardzo skomplikowane obliczenia, ale generującymi też ruchome obrazy czy komponującymi muzykę.

Historia komputera elektronicznego zaczyna się podczas II wojny światowej. W roku 1941 w Niemczech powstał elektromechaniczny komputer Z3, a w roku 1945 w USA – komputer ENIAC oparty na lampach elektronowych, który do niedawna uważany był za pierwszy komputer. W roku 1948 amerykański matematyk, John von Neumann, opracował koncepcję logiczną i zaproponował architekturę komputerów stosowaną do dziś. Wynalazek tranzystora (1948 rok) i jego masowa produkcja (rozpoczęta w roku 1954) pozwoliły na znaczną miniaturyzację kolejnych maszyn. Na przerzutnikach

tranzystorowych budowane były pierwsze polskie komputery z serii Odra. Kolejny krok postępu technologicznego to opracowanie pierwszych układów scalonych. W projekcie należy uwzględnić pojawienie się i rozwój komputerów PC i Mac, urządzeń mobilnych, systemów operacyjnych oraz najnowsze pomysły na komputery.

Ponieważ komputer wykonuje programy, należy też omówić powstawanie języków programowania wyższego rzędu (pierwszym językiem był Fortran, który powstał w roku 1954).

Zadania szczegółowe

1. Historia maszyn liczących i pierwszych komputerów.
2. Historia i rozwój komputerów PC i Mac.
3. Najnowsze pomysły na komputery, w tym urządzenia mobilne.
4. Rozwój języków programowania.
5. Rozwój oprogramowania użytkowego.
6. Przetwarzanie danych w chmurze obliczeniowej.
7. Wpływ pojawienia się Internetu na rozwój technologii informacyjno-komunikacyjnych.

Uwagi dotyczące realizacji projektu

- Projekt jest rozbudowany, dlatego warto wybrać tylko najistotniejsze fakty dotyczące komputerów, oprogramowania i Internetu, a następnie połączyć wszystkie zadania, zapisując je w postaci prezentacji multimedialnej (jednej lub trzech).
- Podczas realizowania zadań 1-5 można skorzystać z opracowania umieszczonego w pliku *Historia i rozwój.doc* (CD).
- Proponowana liczba osób w grupie: 7.

4. Szanse i zagrożenia związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych

Projekt 5. Szanse związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych

Celem projektu jest wyjaśnienie szans dla rozwoju społecznego i gospodarczego oraz dla obywateli, związanych z rozwojem informatyki i TIK.

Należy przedstawić zastosowania informatyki w różnych dziedzinach (praca, nauka, życie codzienne, usługi), docenić aspekty społeczne rozwoju i zastosowań informatyki.

W projekcie należy pokazać, w jaki sposób rozwój e-usług wpłynął na kształtowanie się obrazu społecznego i gospodarczego kraju i jakie dał szanse społeczeństwu. Należy zwrócić szczególną uwagę na pomoc, jaką rozwój TIK przyniósł w nauce i na rynku pracy – dzięki rozwojowi różnych form kształcenia na odległość i e-pracy.

Rozwój informatyki i technologii informacyjno-komunikacyjnych zmienił też w wielu przypadkach standard życia osób niepełnosprawnych, m.in. umożliwił zdobycie pracy, a przede wszystkim ułatwił kontakt z innymi ludźmi.

Wykorzystując opracowanie projektu 4, można spróbować poszukać odpowiedzi na pytanie, w jakim zakresie wydarzenia z historii informatyki (i które z nich) miały wpływ na rozwój społeczeństwa informacyjnego.

Zadania szczegółowe

1. Rozwój zastosowań informatyki.
2. Rozwój e-usług (e-banki, e-zakupy, e-aukcje).
3. Kształcenie i praca na odległość.
4. Szanse dla osób niepełnosprawnych.

Uwagi dotyczące realizacji projektu

- Ze względu na bardzo rozległą tematykę zadania 1. należy wybrać tylko kilka dziedzin, skupiając się na kilku wybranych przykładach. Na przykład w przypadku zastosowań informatyki w służbie zdrowia można pokazać, w jaki sposób rozwój technologii wpłynął na możliwości diagnozy stanu zdrowia pacjentów.
- Każde z zadań szczegółowych powinno być przedstawione w kontekście wpływu na zdobywanie, przechowywanie, przetwarzanie i przesyłanie informacji, a także nieograniczone czasem i przestrzenią komunikowanie się między ludźmi.
- Na slajdach prezentacji można wymienić wybrane wydarzenia z rozwoju informatyki i TIK oraz opatrzyć je komentarzem dotyczącym społecznych korzyści, jakie przyniosły kolejne innowacje. Prezentację należy urozmaicić rysunkami (zdjęciami, schematami).
- Proponowana liczba osób w grupie: 4-6.

Projekt 6. Zagrożenia związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych

Celem projektu jest omówienie zagrożeń dla rozwoju społecznego i gospodarczego oraz dla obywateli, związanych z rozwojem informatyki i TIK.

Analizując problem, należy ocenić zagrożenia i ograniczenia, z jakimi możemy się spotkać, korzystając z technologii informacyjno-komunikacyjnych. Szczególną uwagę należy zwrócić na problem uzależnienia się od komputera i Internetu.

W związku z bardzo dynamicznym rozwojem informatyki i TIK, trzeba zwrócić uwagę na pojawiające się nowe zagrożenia.

Zadania szczegółowe

1. Uzależnienie od komputera i Internetu.
2. Kontakty między ludźmi – zanik związków międzyludzkich „twarzą w twarz” i anonimowość kontaktów.
3. Zagrożenia prywatności.
4. Cyberprzemoc.
5. Niewiarygodność informacji.

Uwagi dotyczące realizacji projektu

- Podczas wykonywania tego projektu można wykorzystać również materiały zgromadzone do przygotowania projektu 3, ponieważ przestępczość komputerowa, stanowiąca zagrożenie, wynika z rozwoju informatyki i TIK.
- Projekty 5. i 6. można połączyć w jeden większy projekt i zrealizować go w postaci debaty ZA i PRZECIW.
- Proponowana liczba osób w grupie: 3-5.

Zadanie dla zainteresowanych

Wybierz (wybierzcie) własny problem do opracowania i wykonaj (wykonajcie) projekt według podanego w temacie schematu.