1 Reprezentacja informacji w komputerze.

Dość częstym nieporozumieniem jest pogląd, że komputery „liczą” a człowiek „myśli”. Dawniej nazywano komputery maszynami cyfrowymi i często podkreślano, że pracują one w układzie dwójkowym, a więc przy pomocy zer i jedynek. Uznaje się, że nasze myślenie przebiega w zupełnie inny sposób, gdyż człowiek korzysta z intuicji i nie dokonuje obliczeń. Ten częsty błąd wynika z pomylenia różnych poziomów rzeczywistości. Układ nerwowy człowieka, a w szczególności mózg, składa się z komórek (neuronów) wysyłających impulsy. Znajomość wzbudzeń wszystkich neuronów w mózgu człowieka nie powie nam bynajmniej, jakie są jego myśli. Podobnie znajomość ciągów zer i jedynek we wnętrzu procesora nie jest interesująca. Są oczywiście zasadnicze różnice w sposobie działania mózgu i komputera, komputer nie jest dobrym modelem działania ludzkiego mózgu, jednak w obu przypadkach mamy do czynienia z przetwarzaniem danych. Mamy tu dwa niezależne poziomy opisu. Na poziomie fizycznej realizacji sprzętowej mówimy o obliczeniach: neurony zliczają impulsy podobnie jak zliczają impulsy elementy półprzewodnikowe. Na poziomie symbolicznym mówimy o przesyłaniu i interpretacji informacji, używamy znaków i słów, które nabierają znaczenia zależnego od kontekstu, w którym się pojawiają.

Czym jest informacja? Wbrew pozorom pojęcie to nie jest wcale tak łatwo zdefiniować. Człowiek posiada pewne wyobrażenia o świecie, ucząc się nabywa nie tylko informację ale i wiedzę. Wiemy na przykład jak prowadzić samochód. Wiedza jest czymś bardziej ogólnym niż informacja. Informacją jest stwierdzenie: maksymalna szybkość osiągana przez ten samochód wynosi 160 km/h. Informacja jest pojęciem dość abstrakcyjnym, gdyż podanie maksymalnej szybkości jako 100 mil/h lub 44.4 m/sek zawiera tę samą informację, chociaż liczby są inne. Liczby te moglibyśmy dodatkowo zapisać w zupełnie inny sposób, alfabetem arabskim lub pismem Brailla, nie zmieniając wcale informacji. Konkretna **reprezentacja informacji** to właśnie **dane**.

Wybór reprezentacji informacji jest bardzo ważny dla wygody przetwarzania danych. Pisząc używamy liter alfabetu łacińskiego - jest to pewien sposób reprezentacji języka naturalnego. Innym sposobem, znacznie mniej wygodnym dla większości czytelników, byłoby użycie alfabetu arabskiego lub cyrylicy. Kilkaset lat temu Wietnamczycy (pod wpływem Francuzów) przeszli z chińskich znaków na alfabet łaciński (z dodatkiem wielu akcentów do liter), w 1920 roku Turcy zrezygnowali z alfabetu arabskiego na rzecz łacińskiego a obecnie wiele republik byłego Związku Radzieckiego porzuca cyrylicę, również na rzecz alfabetu łacińskiego. Wygodnie jest używać tego samego zestawu

znaków dla zapisania dźwięków różnych języków. Jest to jednak kwestia umowna, jak szybko przekonuje się każdy podróżnik po Azji.

Reprezentacja liczb przy pomocy znaków rzymskich jest dużo mniej wygodna niż przy pomocy używanych obecnie znaków. Nie wystarczy wiedzieć, jak mnoży się 2 liczby przez siebie na papierze - liczbami zapisanymi przy pomocy znaków rzymskich nie da się tak łatwo operować. Czasami dana reprezentacja jest wygodna z jednego punktu widzenia a niewygodna z innego. Pismo chińskie, czyli reprezentacja języka naturalnego na papierze, jest bardzo wygodne, gdyż zawarta w znakach chińskich informacja jest zrozumiała dla osób mówiących tysiącami dialektów, wymawiającymi ją w bardzo różny sposób. Z drugiej strony nie jest to zapis alfabetyczny i z punktu widzenia tworzenia słownika jest znacznie mniej wygodny, niż stosowany w większej części świata zapis zbliżony do fonetycznego, gdyż nie ma naturalnego sposobu uporządkowania ideogramów. Zapis fonetyczny, a więc zapis dźwięków, a nie idei, nie jest za to zrozumiały dla osób mówiących innymi językami, pomimo wspólnego alfabetu.

Wewnętrzna reprezentacja danych w komputerze nie jest dla większości użytkowników istotna tak jak nie jest dla nas istotna reprezentacja myśli w mózgu innego człowieka. Wyjątkiem są specjaliści od kompresji danych lub obliczeń numerycznych, którzy interesują się wewnętrzną reprezentacją danych by zwiększyć efektywność pisanych przez siebie programów. Dla użytkownika komputera istotne są typy danych i zbiór znaków, jakimi manipulować może program, który je wykorzystuje. Typy danych mogą być różne: odpowiedzi „tak” lub „nie” uważać można za **dane typu logicznego**; teksty to **dane alfanumeryczne** (alfabet+liczby) a liczby, na których wykonywać można operacje arytmetyczne to **dane numeryczne**. Możliwych jest jeszcze wiele innych typów danych, np. **dane graficzne**, dane alfanumeryczne o ustalonej strukturze (**rekordy**), **dane muzyczne** itd.

Reprezentacja danych tekstowych wymaga ustalenia jakiegoś zbioru znaków wspólnego dla różnych komputerów, czyli alfabetu rozszerzonego o cyfry i znaki specjalne.

1.1 Bity i bajty

Słowo „bit” jest skrótem dwóch słów: binary unit, czyli jednostka dwójkowa. Jest to najmniejsza jednostka informacji, pozwalająca odróżnić 2 sytuacje: tak lub nie, 0 lub 1, lewo lub prawo. Wybór jednej z takich możliwości daje nam jeden bit informacji.

Bit to niewiele informacji, ciąg bitów wystarczy jednak, by przekazać dowolną wiadomość. Afrykańskie tam-tamy i europejski telegraf posługują się tylko dwoma znakami: wysoki-niski lub krótki-długi. W życiu codziennym posługujemy się wieloma znakami. Alfabet polski ma 35 liter. Jeśli odróżnimy małe litery od dużych, dodamy do

tego znaki przystankowe i kilka znaków specjalnych otrzymamy prawie 100 znaków. Na klawiaturze komputera znajduje się od 80 do ponad 100 klawiszy, a niektórym klawiszom odpowiada kilka znaków lub funkcji. W sumie lepiej jest mieć nieco więcej możliwości niż 100. Jeśli zbierzemy razem 8 bitów to możemy przy ich pomocy odróżnić 256 znaków. Ciągi 8 bitów nazywa się **bajtami**.

Najbardziej rozpowszechniony standard reprezentowania znaków alfanumerycznych w komputerze został ustalony w Stanach Zjednoczonych i nazywa się go **standardem ASCII** (American Standard Code for Information Exchange, czyli Amerykański Kod Standardowy dla Wymiany Informacji). Popularny jest również standard **ANSI**, używany w niektórych programach pracujących na komputerach osobistych pod kontrolą MS-Windows. Innym, rzadziej spotykanym sposobem reprezentacji znaków jest **EBCDIC** (Extended Binary-Coded-Decimal Interchange Code czyli Rozszerzony Dziesiętny-Dwójkowo-Zakodowany Kod Wymiany), używany przez wiele komputerów centralnych. Standardy te każdemu znakowi przyporządkowują liczbę od 0 do 255, odpowiadającą jej kolejności w uporządkowanym zbiorze znaków, należących do danego standardu. Istnienie różnych standardów powoduje, że wymiana informacji pomiędzy niektórymi komputerami musi być poprzedzona odpowiednim „tłumaczeniem" z jednego systemu kodowania na drugi. Wielu użytkowników komputerów nie zdaje sobie sprawy z istnienia innych standardów niż ASCII i dopiero kłopoty z automatycznym tłumaczeniem znaków przy przesyłaniu danych z jednego komputera na drugi im to uświadamia. Standard ASCII ma obecnie największe znaczenie gdyż używany jest przez komputery osobiste, stacje robocze i niektóre komputery centralne.

Standard ASCII dotyczy podstawowych znaków alfanumerycznych i ustala tylko pierwsze 128 znaków. **Rozszerzony standard ASCII** określa wszystkie 256 znaków, jednak nie w każdym kraju wszystkie te znaki się przydają. Stąd powstały wariacje wokół amerykańskiego rozszerzonego standardu ASCII, zwane „stronami kodowymi", w których mniej potrzebne znaki (o numerach powyżej 127) zastąpiono w różnych wersjach znakami specjalnymi, używanymi przez różne narody, posługujące się rozszerzonym alfabetem łacińskim. Znaki polskie znalazły się na stronie kodowej określanej nazwą „Latin 2", razem z innymi znakami narodowymi krajów Europy Centralnej. Niestety wśród pierwszych 128 znaków w różnych krajach też występują drobne różnice. Powiemy na ten temat więcej przy okazji omawiania drukarek.

Pierwsze 32 znaki standardu ASCII zarezerwowano dla celów specjalnych, reprezentują one kody kontrolne dla drukarek i ekranu. Pozostałe znaki można drukować. Zostały one uporządkowane w taki sposób, że w kodzie dwójkowym, odpowiadającym ich kolejnemu numerowi, od razu możemy rozpoznać, czy mamy do czynienia z cyfrą, czy z literą małą czy dużą. Nikt już nie pamięta o kodzie dwójkowym więc nie ma to praktycznego znaczenia. Kolejność znaków ASCII w stronie standardowej i stronie Latin 2 znaleźć można na końcu tej książki.

System dwójkowy

Wyobraźmy sobie, że mamy do dyspozycji tylko 2 cyfry, 0 i 1. Jak można zapisać wówczas liczbę 2? Podobnie jak robimy to z liczbą 10 w układzie dziesiętnym, którym posługujemy się na codzień. Nie mając cyfry 10 musimy używać dwóch cyfr, 1 i 0. Podobnie postąpimy i w układzie dwójkowym, ale chociaż zapis będzie taki sam, 10, to będzie on reprezentować liczbę 2, pierwszą, dla której brak nam odpowiedniej cyfry. Liczba 3 reprezentowana będzie przez 11 a 4 przez 100. Jeśli mamy 1 a potem np. 5 zer, czyli 100000, to w układzie dziesiętnym oznacza to liczbę $2^5=32$.

System szesnastkowy

Jeśli mamy do dyspozycji cztery bity to daje 16 możliwości, od 0000, 0001, 0010, 0011,... aż do 1111. Te 16 możliwości można zapisać przy pomocy cyfr

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

gdzie zamiast 10 piszemy A i traktujemy to jako nową cyfrę. Możemy więc reprezentować liczby nie przy pomocy dziesięciu cyfr od 0 do 9 ale 16 cyfr, od 0 do F. Jest to reprezentacja szesnastkowa. Gdybyśmy mieli 16 palców wydawałaby się nam pewnie całkiem naturalna ale trudniej byłoby się nam nauczyć tabliczki dodawania czy mnożenia. Jeden bajt składa się z ośmiu bitów a więc dwóch grup po cztery bity. Możemy więc numerować wszystkie 256 bajtów nie od 0 do 255 ale od 0 do FF.

Najnowszym standardem kodowania znaków, ustalonym w 1992 roku, jest Unicode (jest to właściwie część standardu ISO 10646). System ten używa dwubajtowej reprezentacji znaków. W ten sposób mamy do dyspozycji nie 256 a $256^2=65536$ znaków, w tym około 3000 znaków definiowalnych przez użytkownika. Można będzie dzięki temu kodować teksty w prawie wszystkich językach świata, nawet w pismach, w których znaki stawia się od prawej do lewej strony czy z góry na dół. Wystarczy również miejsca na wiele znaków graficznych (znaki chińskie, japońskie i koreańskie). Zamiast zapisywać takie znaki w postaci grafiki (co zajmuje dużo miejsca w pamięci) wystarczy podać 2 bajty. Najnowsze systemy operacyjne, takie jak Windows NT, posługiwać się mają reprezentacją Unicode. Teksty pisane w językach europejskich zajmują jednak przy takiej reprezentacji dwa razy więcej pamięci.

Inna ciekawa propozycja rozwiązania problemu znaków specjalnych nie mieszczących się w standardzie ASCII polegała na stosowaniu kodu o zmiennej długości, tj. liczba bajtów reprezentujących znak byłaby nieustalona. Można to zrobić na kilka sposobów, np. pierwszy bit danego bajtu można uznać za „bit kontynuacji”; jeśli równy jest 0 to jest

Potęgi dwójki

Ile mamy różnych liczb binarnych dla liczb 2-cyfrowych? To proste: $2 \cdot 2 = 2^2 = 4$.
Podobnie dla liczb 4-cyfrowych mamy $2^4 = 16$ możliwości a dla liczb o większej liczbie cyfr odpowiednio $2^8 = 256$, $2^{10} = 1024 = 1K$.
Dla wygody 2^{10}, równe prawie dokładnie 1000, oznacza się jako 1K, czyli kilo, tak jak w kilometrze, który ma tysiąc metrów, czy w kilogramie, który ma tysiąc gramów. W kilobajcie są więc 1024 bajty.
Wyższe potęgi dwójki możemy wówczas oznaczyć jako $2^{16} = 64K = 65536$, $2^{20} = 1024K = 1M$, gdzie znowu stosujemy skrót 1M czyli „jeden mega", zamiast miliona a dokładniej $1024 \times 1024 = 1048576$.
Będziemy również stosować skrót 1G czyli „jeden giga".
$2^{24} = 16M$, $2^{30} = 1024M = 1G$, $2^{32} = 4096M = 4G$.

to nowy znak, a jeśli 1 to należy go traktować łącznie z bajtem poprzednim. Taka reprezentacja umożliwia przechowywanie dowolnej liczby znaków (nawet 65 tysięcy znaków może okazać się zbyt małą liczbą jeśli uwzględnić wszystkie znaki chińskie, koreańskie i japońskie), nie została jednak przyjęta jako standard.

1.2 Systemy liczenia

Obecnie większość ludzi licząc posługuje się dziesiętnym systemem liczenia. Jednak jeszcze niedawno Anglicy posługiwali się systemem monetarnym dwunastkowo-dwudziestkowym, a starsi ludzie liczą jeszcze czasem w tuzinach zamiast w dziesiątkach. Do tej pory amerykanie dzielą stopy na 12 cali, rok dzielimy na 12 miesięcy a dzień na dwa okresy po 12 godzin. Używanie innej reprezentacji liczb niż dziesiętna nie jest więc wcale takie nienaturalne, jak mogłoby się wydawać. Wszystkie obecnie używane systemy opierają się na tym, że wartość danej cyfry określona jest przez pozycję, na której się ona znajduje. Tak więc 24 może oznaczać w systemie dziesiętnym $2 \cdot 10 + 4$, a w systemie dwunastkowym 2 tuziny plus 4 czyli $2 \cdot 12 + 4$. W systemie rzymskim pozycja cyfry nie ma takiego znaczenia, np. XXX zawiera trzy X na różnych pozycjach. Pozycyjne systemy liczenia stały się możliwe dopiero po wprowadzeniu zera i wyparły całkowicie systemy niepozycyjne, takie jak system rzymski czy chiński.

W matematyce rozważa się systemy liczenia o dowolnej podstawie. Z przyczyn technicznych przy projektowaniu i konstruowaniu komputerów używa się systemu najprostszego, jakim jest system dwójkowy. System ten odkryty został przez Leibniza, który interpretował go w sposób mistyczny: zero oznaczało pustkę przed stworzeniem, a jedynka oznaczała Boga. Zerami i jedynkami wyrazić można całą informację.

Czasami wygodnie jest stosować bardzo szczególne systemy liczenia, np. oparte na liczbach pierwszych lub na zmiennej podstawie. Systemy takie wykorzystywane są w komputerach o eksperymentalnej architekturze, np. wyspecjalizowanych w bardzo szybkim mnożeniu ale za to mających trudności z szybkim dodawaniem. Jest to jednak problem interesujący tylko dla specjalistów zajmujących się konstrukcją komputerów przeznaczonych do szczególnych celów.

1.3 Wielkość danych.

Dane przechowywane są w pamięci komputera w postaci zbioru bajtów. Czasami używa się też pojęcia **słowo**, na określenie takiej liczby bitów, na których komputer dokonać może jednocześnie podstawowych operacji. Słowa w spotykanych obecnie komputerach składają się najczęściej z 8, 16, 32, 48 lub 64 bitów. Do zapisania danych w pamięci potrzeba określonej liczby bajtów lub słów. Liczba ta nazywa się wielkością zbioru danych. Dla większych zbiorów danych, przechowywanych w **plikach** na dyskach lub innych nośnikach, podaje się ile tysięcy lub milionów bajtów zawierają. Ściślej rzecz biorąc przyjęło się posługiwać nie tyle tysiącami bajtów, co wielokrotnościami dziesiątej potęgi dwójki, tj. 2^{10}=1024, nazywanymi kilobajtami. Jeden kilobajt to 1024 bajty (patrz ramka).

Wielkości zbiorów danych będziemy więc wyrażać w kilobajtach (KB) lub megabajtach (MB). Tylko w bardzo dużych archiwach osiągnąć one mogą rozmiary gigabajtów (GB). Istnieje również jednostka 1024 razy większa, zwana terabajtem (TB). Jest to jednostka bardzo duża, ale nie astronomicznie wielka - w Bibliotece Kongresu USA, jednej z największych bibliotek świata, zapisanych jest około 20 TB informacji. Jeszcze większą jednostką jest petabajt (PB) równy 1024 terabajty. Dane zbierane w niektórych eksperymentach naukowych mogą przyjmować takie monstrualne rozmiary.

Czasami podaje się wielkości zbiorów danych w kilobitach (Kb) lub megabitach (Mb), ale nawet w pismach komputerowych redakcja nie przestrzega reguły pisania skrótu bajtów i bitów odpowiednio jako dużego B i małego b. Często też mówi się o zamiennie o plikach i zbiorach danych, chociaż pojęcie zbioru jest o wiele szersze niż pojęcie pliku.

1.4 Typy danych

Uporządkowany zbiór bajtów, np. tekst listu lub zbiór liczb, można nazwać i zapamiętać w postaci **pliku**. Pliki zawierać mogą dane różnych typów. Mogą to być dane tekstowe, np. słowo „tysiąc", mogą to być dane numeryczne, np. 1000. W obu przypadkach informacja jest ta sama, jednak jej sposób zapisu i wykorzystania różny. Na danych numerycznych można wykonywać **operacje arytmetyczne**: dodawać, mnożyć itp. Dane tekstowe można co najwyżej **uporządkować** według jakiegoś klucza (np. według alfabetu), łączyć lub je ze sobą porównywać. Nie można pomnożyć tekstu przez liczbę gdyż nie są to dane tego samego typu. Trzecią, podstawową operacją jaką wykonać można na danych jest ich **przesunięcie** z jednego miejsca na drugie. Wszystkie wykonywane przez komputer czynności składają się z tych trzech podstawowych operacji: arytmetycznych, porównania i przesunięcia.

W oparciu o dane tekstowe i numeryczne utworzyć można bardziej złożone struktury. Dane numeryczne mogą reprezentować struktury matematyczne typu wektorów, macierzy, czy wielowymiarowych tablic. Informacja tekstowa może składać się z grup znaków o ustalonej strukturze, czyli **rekordów**. Przykładem rekordu może być adres, zapisany na kopercie listu: składa się on z nazwiska, ulicy, numeru domu, miasta, kodu pocztowego.

Informatycy, jak każda grupa specjalistów, wymyślili lub częściowo przejęli od matematyków specjalistyczny żargon odstraszający laików. Mówią np. o „konkatenacji łańcuchów znakowych" mając na myśli połączenie dwóch fragmentów tekstu w jedną całość. Nie tak dawno temu podręczniki opisujące oprogramowanie dla dużych systemów komputerowych pisane były okropnym żargonem. Na szczęście masowe rozpowszechnienie się komputerów wymusiło na specjalistach tworzenie przyjaznego oprogramowania i bardziej zrozumiałej dokumentacji.