



Temat:

Binarny system liczbowy cz.3

Dzielenie całkowite i reszta z dzielenia

Program *Przeliczanie temperatury* z tematu 5 zawierał operację dzielenia na liczbach całkowitych. Wynik tego działania był obcięty do części całkowitej według zasady: jeśli dzielna i dzielnik są typu **int**, to iloraz też musi być typu **int**. Jeśli więc użyjemy deklaracji **int** `iloraz = 123/2;`, zmiennej `iloraz` zostanie przypisana wartość 61. Ta wiedza przyda nam się podczas zapisywania kroku algorytmu zamiany, w którym dzielimy ostatnio wyznaczony iloraz przez 2.

Do wyznaczania reszty z dzielenia służy operator **%**, zwany **modulo**. Na przykład wartością zmiennej `reszta` po deklaracji i przypisaniu **int** `reszta = 123%2;` będzie 1.

Program *Przeliczanie temperatury*,
s. 104 

• **Modulo** to operacja wyznaczania reszty z dzielenia jednej liczby przez drugą.

Ćwiczenie 4

Sprawdź efekt działania programu, którego kod wewnątrz funkcji `main` ma postać:

```
int iloraz, reszta;
iloraz = 123/2;
reszta = 123%2;
cout<<iloraz<<endl;
cout<<reszta<<endl;
```

Warto wiedzieć

Reszta z dzielenia dowolnej liczby całkowitej przez liczbę k może przyjąć wartości od 0 do $k-1$.

Program oblicza iloraz i resztę po wprowadzeniu liczby całkowitej

```
iloraz_reszta1.cpp x
1      #include <iostream>
2
3      using namespace std;
4
5      int main()
6      {
7          int liczba, iloraz, reszta;
8          cout<<"Wprowadz liczbe "<< endl;
9          cin>>liczba;
10         iloraz=liczba/2;
11         reszta=liczba%2;
12         cout<<iloraz<<endl;
13         cout<<reszta<<endl;
14         return 0;
15     }
16
```

Program oblicza iloraz i resztę po wprowadzeniu liczby całkowitej

Kompilacja

CTRL + F9

Uruchomienie

CTRL + F10

 C:\Users\Nauczyciel\Desktop\C++\iloraz_reszta1.exe

Wprowadz liczbe

123

61

1

Process returned 0 (0x0) execution time : 1.500 s

Press any key to continue.

Uruchomienie programu

Filmik – Tablice w języku C++

Filmik do pobrania:

- strona internetowa www.zstw.pl
- Menu górne/Dla ucznia/Pobierz plik/KLASA II

Tablica – struktura danych

Do przechowywania określonej liczby elementów tego samego typu można posłużyć się strukturą danych nazywaną **tablicą** (ang. *array*). 📌 Tablica

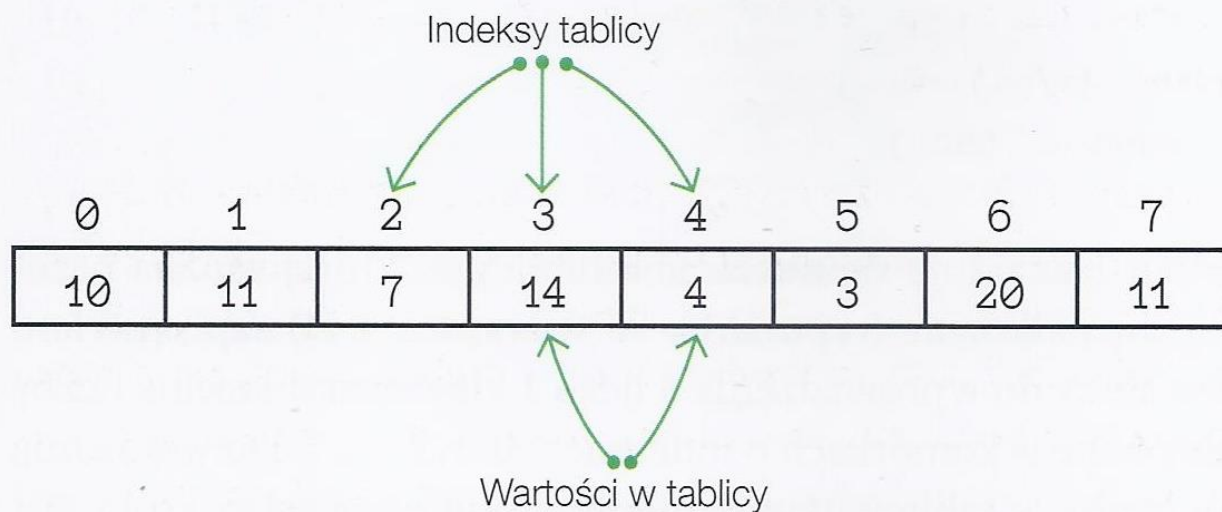
Aby korzystać z tablicy, trzeba ją wcześniej **zadeklarować**, czyli określić jej **nazwę, rozmiar i typ** przechowywanych informacji. Na przykład tablicę o nazwie `tab`, w której będziemy przechowywać wartości ośmiu liczb całkowitych, zadeklarujemy w C++ następująco:

```
int tab[8];
```

Tablicę można porównać do jednowierszowej tabeli, w której komórkach zapisujemy kolejne liczby. Każda komórka tablicy ma numer, zwany **indeksem**. Komórki numeruje się od 0 (rys. 7.10). 📌 Indeks

👍 Dobra rada

10-elementową tablicę liczb rzeczywistych (typu **float**) możesz zadeklarować następująco:
`float tablica[10];`



Rys. 7.10. Interpretacja graficzna tablicy 8-elementowej

Odwołanie się do jakiegoś elementu tablicy

Aby odwołać się do jakiegoś elementu tablicy, wystarczy podać jego współrzędną (indeks). Instrukcje:

```
tab[0] = 10;
```

```
tab[7] = 11;
```

powodują przypisanie wartości 10 pierwszemu elementowi tablicy (o indeksie 0) i przypisanie wartości 11 ostatniemu elementowi (o indeksie 7).

Nadawanie wartości początkowych tablicy

W momencie deklarowania tablicy można określić wartości początkowe jej elementów. Można to zrobić na różne sposoby:

1. `int a[4] = {0};`
2. `int b[4] = {86, 12, 24, 1000};`
3. `int c[4] = {3, 5};`

Czteroelementowa tablica o nazwie `a` z wiersza 1 będzie wypełniona zerami, tablica `b` z wiersza drugiego będzie przechowywała wartości 86, 12, 24 i 1000. Dwa pierwsze elementy tablicy `c` z wiersza trzeciego będą przechowywały odpowiednio wartości 3 i 5, a pozostałe elementy – będą przechowywały wartość 0.

Tablice – wczytywanie i wypisywanie danych

Napišemy krótki program o nazwie *Cyfry wspak*, który po wczytaniu z klawiatury ośmiu liczb jednocyfrowych wyświetli te liczby w odwróconej kolejności. Do wykonania operacji wczytywania i wypisywania danych użyjemy instrukcji **for**.

Zasadniczy fragment programu, czyli kod znajdujący się wewnątrz funkcji `main`, będzie miał postać:

```
1. int tab[8];
2. for(int i=0; i<=7; i++)
3.     cin >> tab[i];
4.
5. for(int j=7; j>=0; j--)
6.     cout << tab[j] << " ";
```

W wierszu 1 znajduje się deklaracja tablicy, w której można przechować liczby całkowite (typu `int`). W wierszach 2 i 3 zapisano kod pętli, która służy do wprowadzenia 8 liczb z klawiatury: kolejne liczby będą zapisywane w komórkach o indeksach: 0, 1, 2, ..., 7. Do wskazania położenia liczby w tablicy użyto zmiennej pomocniczej `i`, która jest jednocześnie licznikiem pętli. Ponieważ liczby mają być przez program wypisane w odwróconej kolejności, w pętli z wierszy 5 i 6 wypisywanie zawartości tablicy rozpoczyna się od elementu o indeksie 7. Do kontroli liczby iteracji i jednocześnie ustalenia pozycji liczby w tablicy służy zmienna pomocnicza `j`.

Kod źródłowy - Program Cyfry wspak

```
tablica1.cpp x
1      #include <iostream>
2
3      using namespace std;
4
5      int main()
6      {
7          int tab[8];
8          for(int i=0; i<=7; i++)
9              cin >> tab[i];
10
11         for(int j=7; j>=0; j--)
12             cout << tab[j] << " ";
13         return 0;
14     }
15
```

Kod źródłowy – Program Cyfry wspak

Kompilacja

CTRL + F9

Uruchomienie

CTRL + F10

 C:\Users\Nauczyciel\Desktop\C++\tablica1.exe

```
12
34
2
77
190
123
56
43
43 56 123 190 77 2 34 12
Process returned 0 (0x0)    execution time : 26.312 s
Press any key to continue.
```

Uruchomienie programu

7.4. Program zamieniający liczbę dziesiętną na binarną

Przygotujemy się teraz do stworzenia programu, który po wpisaniu z klawiatury liczby naturalnej, nie większej niż 255, wypisze ciąg zer i jedynek przedstawiający binarną reprezentację komputerową tej liczby. Ograniczymy się do liczb, które da się zapisać w jednym **bajcie** (ang. *byte*), czyli jako liczby binarne co najwyżej ośmiocyfrowe, złożone z ośmiu bitów.

Zapiszmy specyfikację problemu algorytmicznego.

Specyfikacja

Dane: liczba całkowita (nieujemna) nie większa niż 255.

Wynik: zapis binarny liczby całkowitej.

Liczba	Iloraz z dzielenia przez 2	Reszta z dzielenia przez 2
123	61	1
61	30	1
30	15	0
15	7	1
7	3	1
3	1	1
1	0	1

Program oblicza iloraz i resztę po wprowadzeniu liczby całkowitej

Aby wyznaczyć postać binarną liczby dziesiętnej, możemy postępować w sposób przedstawiony w tabeli 7.2 ze s. 133. Wystarczy zapamiętywać kolejne ilorazy z dzielenia przez 2 oraz otrzymywane reszty z dzielenia. Dokładny zapis działania algorytmu przedstawia poniższa lista kroków.

- ➊ Podziel liczbę przez 2, zapisz resztę z dzielenia i zapamiętaj iloraz (część całkowitą).
- ➋ Dopóki ostatnio otrzymany iloraz jest większy od 0, powtarzaj kroki ➌, ➍ i ➎.
- ➌ Wyznacz resztę z dzielenia ostatnio zapamiętanego ilorazu przez 2.
- ➍ Zapisz resztę z dzielenia przed ostatnią wcześniej zapisaną resztą.
- ➎ Podziel iloraz przez 2 i zapamiętaj część całkowitą wyniku jako nowy iloraz.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int liczba, iloraz, cyfry[8]={0};
8
9      cout << "Podaj liczbe: "; cin >> liczba;
10
11     iloraz = liczba/2;
12     cyfry[7] = liczba% 2;
13
14     int i=1;
15     while(iloraz>0)
16     {
17         cyfry[7-i] = iloraz% 2;
18         iloraz = iloraz/ 2;
19         i++;
20     }
21
22     cout << "Zapis binarny liczby " << liczba << " to: ";
23     for (int i=0; i<=7; i++)
24         cout << cyfry[i];
25
26     cout << endl;
27
28     return 0;
29 }
```

Kod źródłowy – Program Zamiana

Kod źródłowy – Program Zamiana

Kompilacja

CTRL + F9

Uruchomienie

CTRL + F10

 C:\Users\Nauczyciel\Desktop\C++\zamiana.exe

Podaj liczbę: 123

Zapis binarny liczby 123 to: 01111011

Process returned 0 (0x0) execution time : 3.410 s

Press any key to continue.

Uruchomienie programu