

# PJC

(ćwiczenia 2)

## Zadanie 1

Napisz zestaw funkcji do badania liczb pierwszych i dzielników liczb.

- **bool isPrime**(int n) sprawdza, czy podana liczba jest pierwsza;
  - **void printDivisors**(int n) wypisuje wszystkie dzielniki podanej liczby naturalnej (łącznie z jedynką i samą liczbą);
  - **int numDivisors**(int n) zwraca liczbę dzielników podanej jako argument liczby naturalnej N (łącznie z jedynką i samą liczbą N);
  - **int gcd**(int m, int n) znajduje największy wspólny dzielnik podanych dwóch liczb używając algorytmu Euklidesa (Księga VII i X Elementów):  
Dla danych  $a; b \in \mathbb{N}$ :
    - Dopóki  $a \neq b$  : większą z liczb  $a; b$  pomniejsz o wartość mniejszej
    - Zwróć  $a$
  - **int phi**(int n) zwraca dla danej liczby naturalnej N ilość wszystkich liczb z zakresu  $[1;N]$  względnie pierwszych z N (dla których największy wspólny dzielnik z N wynosi 1). Jest to słynna funkcja Eulera. Na przykład  $\phi(10) = 4$ , bo tyle jest liczb naturalnych mniejszych od 10 i względnie pierwszych z 10 (są to liczby 1, 3, 7 i 9).
- 

## Zadanie 2

Napisz funkcję o prototypie

```
const double* aver(const double* arr, size_t size, double& average);
```

która pobiera tablice liczb typu double i jej wymiar oraz zmienna average typu double przez referencje. Zadaniem funkcji jest wstawienie do zmiennej average średniej arytmetycznej elementów tablicy oraz zwrócenie wskaźnika do tego elementu tablicy, którego wartość jest najbliższa tej średniej.

Na przykład następujący program

```
#include <iostream>
#include <cmath>
const double* aver(const double* arr, size_t size, double& average) {
    // ...
}
int main () {
    using std::cout; using std::endl;
    double arr[] = {1,7,5,4,3,2};
    size_t size = sizeof(arr)/sizeof(arr[0]);
    double average = 0;
    const double* p = aver(arr, size, average);
    cout << *p << " " << average << endl;
}
```

powinien wypisać

4 3.66667

---

### Zadanie 3

Napisz funkcję pobierającą tablice nieujemnych liczb całkowitych (i jej wymiar) i wypisującą na ekranie, za pomocą symbolu „\*”, „histogram” dla danych z tablicy, to znaczy „słupki”, wyrównane od dołu, o wysokościach równych wartościom kolejnych elementów tablicy.

Na przykład program

```
#include <iostream>
using namespace std;
void histo(int arr[], size_t size) {
    // ...
}
int main() {
    int arr[]{2,1,0,7,1,9};
    size_t size = sizeof(arr)/sizeof(*arr);
    histo(arr,size);
}
```

powinien wydrukować:

```
      *
      *
    * *
    * *
    * *
    * *
    * *
    * *
  * * *
** ****
```

---