

# Wstęp do programowania

## Zestaw 3 — wyrażenia i instrukcje

**Zadanie 1.** Napisać program, za pomocą którego wczytasz dwie liczby całkowite a następnie wyznacysz największy wspólny dzielnik (NWD) za pomocą Algorytmu Euklidesa.

Praktycznym i szybkim sposobem obliczania największego wspólnego dzielnika dwóch liczb całkowitych jest algorytm Euklidesa. Jest to jeden z najstarszych algorytmów - został opisany przez Euklidesa ok. roku 300 p.n.e. Opiera się on na spostrzeżeniu, że jeśli od większej liczby odejmiesz mniejszą, to mniejsza liczba i otrzymana różnica będą miały taki sam największy wspólny dzielnik jak pierwotne liczby. Jeśli w wyniku kolejnego odejmowania otrzymasz parę równych liczb, oznacza to, że znalazłeś NWD.

Źródło: <http://www.math.edu.pl/algorytm-euklidesa>

**Zadanie 2.** Napisz program, w którym zostanie wypisany tekst np. "Hello World!". Powtórz wypisywanie tekstu. Powtarzanie zorganizuj w następujący sposób:

- (a) powtórz wypisywanie tekstu  $n$  razy,  $n$  podaje użytkownik programu,
- (b) powtarzanie wypisywania tekstu uzależnij od decyzji użytkownika programu.

**Zadanie 3.** Napisz program, który powtarza wczytywanie liczby rzeczywistej typu `double` dopóki liczba jest różna od zera. Jeśli wczytana liczba jest większa od zera, to program oblicza i wyświetla jej pierwiastek kwadratowy, natomiast jeśli wczytana liczba jest mniejsza od zera, to program oblicza i wyświetla jej kwadrat. Jeśli liczba jest równa zero, to program kończy działanie, w przeciwnym wypadku prosi o wprowadzenie kolejnej liczby, itd. Dodatkowo, w programie policz ile liczb zostało wczytanych oraz ile wśród nich było liczb ujemnych a ile dodatnich.

Przykład działania programu:

```
Podaj liczby. Wartosc zero konczy wczytywanie.
```

```
Podaj liczbe: 3
```

```
Pierwiastek: 1.73205
```

```
Podaj liczbe: -4
```

```
Kwadrat: 16
```

```
Podaj liczbe: 0
```

```
KONIEC WCZYTYWANIA
```

```
Liczba wczytanych wartosci: 3
```

```
w tym dodatnich: 1
```

```
w tym ujemnych: 1
```

**Zadanie 4.** Napisać program, który pobierający od użytkownika liczby całkowite aż do podania liczby 0 oraz wyznaczający i wypisujący, ile liczb różnych od zera zostało wczytanych oraz ile zostało wczytanych liczb dodatnich a ile liczb ujemnych. Program dodatkowo ma wypisywać informację, że wczytana liczba jest dodatnia lub, że wczytana liczba jest ujemna.

Przykład działania programu:

Podaj liczby. Wartość zero kończy wczytywanie.

```
1: 5
liczba dodatnia
2: -3
liczba ujemna
3: 4
liczba dodatnia
4: 4
liczba dodatnia
5: -2
liczb ujemna
6: 0
Liczba wczytanych liczb różnych od zera: 5
w tym dodatnich: 3
w tym ujemnych: 2
```

**Zadanie 5.** Rozbuduj program z zadania 3 lub 4 dodając do niego fragmenty odpowiedzialne za policzenie sumy oraz średniej liczb ujemnych oraz sumy i średniej liczb dodatnich. Wypisz otrzymane wartości i ewentualnie odpowiednie komentarze.

**Zadanie 6.** Napisać program pobierający od użytkownika całkowitą dodatnią liczbę  $M$ , a następnie wyznaczający ile kolejnych liczb całkowitych dodatnich należy dodać, aby otrzymać najmniejszą liczbę nie mniejszą niż  $M$ .

**Zadanie 7.** Napisać program pobierający od użytkownika dwie liczby rzeczywiste oznaczające odpowiednio długość trasy w km i wartość w litrach oznaczającą zużyte na tym odcinku drogi paliwo i zwracający średnie zużycie paliwa na 100 km. Dodaj do tego programu możliwość powtarzania obliczeń bez konieczności uruchamiania go jeszcze raz. Powtarzanie zorganizuj w następujący sposób:

- (a) powtórz obliczenia  $n$  razy, wartość  $n$  podaje użytkownik programu,
- (b) powtarzanie obliczeń uzależnij od decyzji użytkownika programu.

**Zadanie 8.** Napisać program rozwiązujący równanie kwadratowe (postaci  $ax^2 + bx + c = 0$ , o ile  $a \neq 0$ ) o podanych przez użytkownika współczynnikach całkowitych. Dodaj do tego programu możliwość powtarzania obliczeń bez konieczności uruchamiania go jeszcze raz.

**Zadanie 9.** Napisać program, który pobiera od użytkownika liczby całkowite aż do podania liczby 0 (niebranej pod uwagę w dalszych obliczeniach) i sprawdzający, czy dla trójek kolejno podawanych liczb  $a, b, c$  spełniony jest warunek  $a < b < c$ . Jeśli warunek jest spełniony program powinien wypisać trójkę. Policzyć i wypisać ile takich trójek wystąpiło.

**Zadanie 10.** Użytkownik podaje pole powierzchni koła i pole powierzchni kwadratu. Program ma sprawdzić i wypisać informację o tym czy kwadrat mieści się w kole? Dodaj do tego programu

możliwość powtarzania obliczeń bez konieczności uruchamiania go jeszcze raz oraz policz ile razy kwadrat zmieścił się w kole.

**Zadanie 11.** Napisać program pobierający od użytkownika dwie liczby całkowite oznaczające licznik i mianownik pewnego ułamka. Jeśli jest to możliwe przedstawić wczytany ułamek w postaci całość i część ułamkowa po skróceniu. Tzn. z ułamka  $32/12$  powinna powstać liczba 2 i  $2/3$ . Dodaj do tego programu możliwość powtarzania obliczeń bez konieczności uruchamiania go jeszcze raz. W programie należy wykorzystać zadanie 1.

**Zadanie 12.** (\*) Napisz program sprawdzający znajomość tabliczki mnożenia. Program generuje pseudolosowo dwie liczby z zakresu  $[1, 10]$ , a następnie wyświetla komunikat, np. „Ile wynosi  $5 * 8$ ?”. Użytkownik podaje odpowiedź, a program wyświetla informację, czy odpowiedź jest prawidłowa, czy nie i ewentualnie podaje prawidłową odpowiedź. Po sprawdzeniu 10 par liczb program wyświetla liczbę prawidłowych odpowiedzi i wystawia ocenę: 10 prawidłowych odpowiedzi - 5.0, 9 - 4.5, 8 - 4.0, 7 - 3.5, 6 - 3.0, 5 i mniej - 2.0. Do pseudolosowego generowania liczb zastosuj funkcje `srand( )` i `rand( )`. Przykład zastosowania funkcji `srand( )` i `rand( )` znajduje się w pliku `losowanie.cpp`.

**Zadanie 13.** (\*) Napisać program, który pobiera od użytkownika liczby całkowite aż do podania liczby ujemnej i wypisuje słownie cyfry dziesiętne liczby całkowitej. Na przykład dla liczby 7034 prawidłowym wynikiem jest `siedem zero trzy cztery`

**Zadanie 14.** (\*) Napisać program, który pobiera od użytkownika liczby całkowite aż do podania liczby niedodatniej i sprawdza czy podana liczba całkowita jest liczbą pierwszą. Dodaj do programu możliwość wypisywania liczb na przykład w postaci:

```
Podaj liczbe dodatnia: 34
34 = 1 * 2 * 17
Podaj liczbe dodatnia 123456
123456 = 1 * 2 * 2 * 2 * 2 * 2 * 2 * 3 * 643
Podaj liczbe dodatnia 643
643 = 1 * 643 (liczba pierwsza)
```