

---

## Zajęcia 2 – instrukcje warunkowe, pętle

---

1. Napisać program realizujący funkcje prostego kalkulatora, pozwalającego na wykonywanie operacji dodawania, odejmowania, mnożenia i dzielenia na dwóch liczbach rzeczywistych. Program ma identyfikować sytuację wprowadzenia błędnego symbolu działania oraz próbę dzielenia przez zero. Zastosować instrukcję `switch` do wykonania odpowiedniego działania w zależności od wprowadzonego symbolu operacji. Scenariusz działania programu:
  - a) Program wyświetla informację o swoim przeznaczeniu.
  - b) Wczytuje pierwszą liczbę.
  - c) Wczytuje symbol operacji arytmetycznej: `+`, `-`, `*`, `/`.
  - d) Wczytuje drugą liczbę.
  - e) Wyświetla wynik lub - w razie konieczności - informację o niemożności wykonania działania.
  - f) Program kończy swoje działanie po naciśnięciu przez użytkownika klawisza Enter.
2. Napisać program pobierający od użytkownika dwie liczby całkowite  $A$  oraz  $B$ ,  $A < B$ , a następnie wyznaczający sumę ciągu liczb  $(A, A + 1, \dots, B)$ . Obliczenia należy wykonać trzykrotnie stosując kolejno pętle: `while`, `do-while`, `for`.

Przykład:

Dla  $A = 4$  i  $B = 11$  program powinien wyświetlić:

```
60
60
60
```

3. Napisać program, który pobiera od użytkownika ciąg liczb całkowitych. Pobieranie danych kończone jest podaniem wartości 0 (nie wliczana do danych). W następnej kolejności program powinien wyświetlić sumę największej oraz najmniejszej z podanych liczb, średnią arytmetyczną wprowadzonych liczb, a także ile z wprowadzonych liczb było jest większych od średniej.

Przykład:

Użytkownik podał ciąg: 1, -4, 2, 17, 0.

Wynik programu:

```
2 // suma min. i maks.  
4.0 // średnia  
1 // ile większych od średniej
```

4. Napisać program działający w trybie konsolowym (tekstowym) i rysujący na ekranie prostokąt. Użytkownik podaje znak wypełnienia prostokąta, pozycję lewego górnego rogu prostokąta  $(x, y)$  oraz długości boków prostokąta  $(ab)$ . Przyjmujemy, że lewy górny narożnik konsoli ma współrzędne  $(x, y) = (1, 1)$ .

Przykład:  $x=6, y=3, a=4, b=6, zn='x'$

```
>  
>  
> _____xxxxxx  
> _____xxxxxx  
> _____xxxxxx  
> _____xxxxxx
```

ozn.

```
> - nowa linia,  
_ - znak spacji.
```

5. Napisać program rysujący w konsoli „choinkę” złożoną ze znaków gwiazdki (\*). Użytkownik programu powinien podać liczbę całkowitą  $n, n > 0$ , określającą wysokość choinki (liczbę wierszy).

Przykład: dla  $n = 5$  wynik powinien wyglądać następująco:

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

6. Napisać program, który pobiera od użytkownika liczbę całkowitą, a następnie:
- oblicza sumę cyfr tej liczby,
  - stosunek średniej arytmetycznej cyfr parzystych do średniej arytmetycznej cyfr nieparzystych.
7. Napisać program, który pobiera od użytkownika liczbę całkowitą nieujemną i sprawdza, czy jej cyfry tworzą palindrom. Przykładowo, cyfry liczby „12321” tworzą palindrom, natomiast cyfry liczby „12345” – nie.
8. Napisać program, który sprawdza, czy podana liczba całkowita  $n, n > 1$ , jest liczbą pierwszą.

9. Napisać program rysujący w konsoli *szachownicę*  $8 \times 8$  o wielkości pola  $n$ ,  $n \in \{1, 2, \dots, 5\}$  podanego przez użytkownika. Szachownica powinna mieć również „obramowanie”, jak w poniższym przykładzie.

Przykład: dla  $n = 1$  wynik powinien wyglądać następująco:

```
+-----+
| # # # # |
|# # # # |
| # # # # |
|# # # # |
| # # # # |
|# # # # |
| # # # # |
|# # # # |
+-----+
```

Dla wartości  $n = 2$  wynik powinien być następujący:

```
+-----+
| ## ## ## ## |
| ## ## ## ## |
|## ## ## ## |
|## ## ## ## |
| ## ## ## ## |
| ## ## ## ## |
|## ## ## ## |
|## ## ## ## |
| ## ## ## ## |
| ## ## ## ## |
|## ## ## ## |
|## ## ## ## |
| ## ## ## ## |
| ## ## ## ## |
|## ## ## ## |
|## ## ## ## |
+-----+
```