Zajęcia 2 – instrukcje warunkowe, pętle

1. Napisać program wyświetlający wszystkie dzielniki dla podanej liczby naturalnej n nie większe od $\sqrt{n}$. Należy również wyświetlić informację, czy liczba jest pierwsza, tj. czy liczba jej dzielników jest równa 2. Uwaga: 1 nie jest liczbą pierwszą.

   Wskazówka. Nie ma potrzeby korzystania z funkcji `sqrt` (ani podobnych).

   Przykład:

   ```
   Podaj n: 1100
   Dzielniki <= sqrt(1100) to: 1 2 4 5 10 11 20 22 25
   Czy pierwsza: nie

   Podaj n: 1
   Dzielniki <= sqrt(1) to: 1
   Czy pierwsza: nie

   Podaj n: 2
   Dzielniki <= sqrt(2) to: 1
   Czy pierwsza: tak
   ```

2. Dany jest następujący algorytm:

   a) Dane wejściowe: liczba całkowita n, $n \geqslant 1$

   b) Powtarzaj dopóki $n \neq 1$

      - jeżeli n jest nieparzyste, to wykonaj $n \leftarrow 3n + 1$
      - w przeciwnym razie $n \leftarrow n/2$

   Napisz program, który dla podanej przez użytkownika liczby całkowitej wypisze na konsoli:

   a) kolejne wartości n uzyskane zgodnie z powyższym algorytmem.

   b) najmniejszą i największą z obliczonych wartości oraz sumę wszystkich kolejnych wartości zmiennej n.

   Przykład dla $n = 42$:

   ```
   Kolejne wart. ciągu: 42 21 64 32 16 8 4 2 1
   Wart. najmniejsza: 1, najwieksza: 64, suma: 190
   ```

3. Napisać program realizujący funkcje prostego kalkulatora, pozwalającego na wykonywanie operacji dodawania, odejmowania, mnożenia i dzielenia dla dwóch liczb rzeczywistych. Program powinien 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 (typ `char`): +, -, *, /.

   d) Wczytuje drugą liczbę.

   e) Wyświetla wynik albo informację o niemożności wykonania działania w przypadku próby dzielenia przez 0.

   f) Program pyta czy zakończyć działanie – wprowadzenie 't' powoduje zakończenie działania, w przeciwnym razie program wraca do pkt. 1.

4. 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:

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

   Wskazówka. W tab. 2.1 pokazano kolejne kroki rozwiązania zadania. Liczba znaków w ostatnim wierszu (szer. choinki) wynosi $2n - 1$.

Tabela 2.1: Kolejne kroki rozwiązania zadania 4 dla $n = 4$.

Krok I	Krok II	Krok III	Krok IV
*******	*******	*	*
	*******	***	***
	*******	*****	*****
	*******	*******	*******

5. Napisać program, który pobiera od użytkownika liczbę całkowitą n, a następnie drukuje na konsoli:

   a) sumę cyfr liczby n;

b) stosunek sumy cyfr parzystych do sumy cyfr nieparzystych, ale tylko gdy liczba ma cyfry nieparzyste;

c) *informację czy ciąg cyfr liczby n jest palindromem (nie ma potrzeby stosowania łańcuchów oraz tablic).

Wskazówka. Kolejne cyfry liczby można uzyskać za pomocą systematycznego dzielenia przez 10 aż do uzyskania 0, sprawdzając „po drodze" reszty z dzielenia przez 10 dla kolejnych wartości. Przykład:

```
Podaj n: 12321
Suma cyfr: 9
Stos. sumy cyfr parzystych do sumy nieparzystych: 0.8
Czy cyfry tworzą palindrom: tak

Podaj n: 244
Suma cyfr: 10
Czy cyfry tworzą palindrom: nie
```

6. Napisać program rysujący w konsoli *szachownicę* 8×8 o wielkości pola n, $n \in \{1, 2, \ldots 5\}$ 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:

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