
Zajęcia 2 – instrukcje warunkowe, pętle

W poniższych zadaniach nie ma potrzeby stosowania tablic – wszelkie operacje na ciągach można wykonać „na bieżąco” bez spamiętywania wszystkich elementów.

1. Napisać program, który pobiera od użytkownika liczbę całkowitą dodatnią, a następnie wyświetla na ekranie kolejno wszystkie liczby nieparzyste nie większe od podanej liczby. Przykład, dla 15 program powinien wyświetlić 1, 3, 5, 7, 9, 11, 13, 15.
2. Napisać program pobierający od użytkownika dwie liczby całkowite A oraz B , $A < B$, a następnie wyznaczający sumę ciągu liczb od A do B , czyli sumę ciągu $(A, A + 1, \dots, B)$. 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 wczytuje od użytkownika liczbę całkowitą dodatnią n , a następnie wyświetla na ekranie wszystkie potęgi liczby 2 nie większe, niż podana liczba. Przykładowo, dla liczby 71 program powinien wyświetlić:

```
1
2
4
8
16
32
64
```

4. Napisać program, który wczytuje liczby podawane przez użytkownika dotąd, dopóki nie podana zostanie liczba 0. Następnie wyświetlić sumę wszystkich podanych liczb.

5. Napisać program, który pobiera od użytkownika ciąg liczb całkowitych. Pobieranie danych kończone jest podaniem wartości 0 (nie jest brana pod uwagę w obliczeniach). W następnej kolejności program powinien wyświetlić:
- najmniejszą i największą podaną wartość,
 - sumę podanych liczb,
 - średnią arytmetyczną podanych liczb.

Przykład:

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

Wynik programu:

```
min: -4
max: 17
suma: 16
średnia: 4
```

6. Gra w „Za dużo, za mało”. Komputer losuje liczbę z zakresu $1 \dots 100$, a gracz (użytkownik) ma za zadanie odgadnąć, co to za liczba poprzez podawanie kolejnych wartości. Jeżeli podana wartość jest:
- większa – wyświetlany jest komunikat „Podałeś za dużą wartość”,
 - mniejsza – wyświetlany jest komunikat „Podałeś za małą wartość”,
 - identyczna z wylosowaną – wyświetlany jest komunikat „Gratulacje” i gra się kończy.
7. 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: $x=6, y=3, a=4, b=6, zn='x'$

```
>
>
> _____xxxxxxx
> _____xxxxxxx
> _____xxxxxxx
> _____xxxxxxx
```

ozn.

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

8. Napisać program, dla podanej liczby całkowitej wyświetla jej dzielniki. Przykładowo, dla liczby 21 dzielniki to: 1, 3, 7, 21.
9. Napisać program, który sprawdza, czy podana liczba całkowita $n, n > 1$, jest liczbą pierwszą.
10. 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.

Wskazówka: do pobierania kolejnych cyfr liczby należy użyć operacji modulo (operator %). Przykładowo, $127 \% 10 = 7$. Kolejną cyfrę uzyskamy dzieląc 127 przez 10 (dzielenie całkowite) i stosując ponownie operację modulo do wyniku, tj. 12.

11. 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:

```

      *
     ***
    *****
   ********
  *********

```

12. Napisać program wspomagający pracę kasjerki/kasjera, który dla podanej kwoty wyświetla liczbę monet poszczególnych rodzajów potrzebną do jej wydania. Program powinien wyświetlać rozwiązanie z minimalną liczbą monet (może być dowolne rozwiązanie z tą samą, minimalną liczbą monet). Dostępne nominały monet to: 5zł, 2zł, 1zł, 50gr, 20gr, 10gr, 5gr, 2gr, 1gr.

Kwota podawana jest w formacie zł [spacja] gr.

Przykładowo, dla danych 12 31 (12zł 31gr) program powinien wyświetlić:

```

2 x 5 zł
1 x 2 zł
1 x 20 gr
1 x 10 gr
1 x 1 gr

```