

# 1 Początki

---

Podróż na tysiąc mil zaczyna się od  
pierwszego kroku

---

Lao-Tse

Zadania w tym zestawie są przeznaczone dla programisty stawiającego pierwsze kroki w języku C. Materiał niezbędny do rozwiązania zadań z tego zestawu zawiera się w podrozdziałach 1–5 oraz 7–8 rozdziału pierwszego [2]. We wszystkich programach w tym zestawie dane pobierane są ze standardowego wejścia i wysyłane na standardowe wyjście.

## **0x00 ZADANIE** ❄

Napisać program wypisujący wartość stałej symbolicznej EOF.

## **0x01 ZADANIE**

Napisać program wyświetlający wartości liczbowe i „efekty działania” znaków specjalnych: `\n`, `\t`, `\r`, `\b`, `\a`, `\f`, `\\`, `\"`.

## **0x02 ZADANIE**

Napisać program, który wykonuje zadaną przez użytkownika operację arytmetyczną w postaci „ $x \square y$ ”, gdzie  $x, y$  są liczbami, zaś  $\square$  jest jednym ze znaków  $+$ ,  $-$ ,  $*$ ,  $/$ .

## **0x03 ZADANIE** ❄

Napisać program wyświetlający na ekranie pierwsze szesnaście potęg dwójki.

## **0x04 ZADANIE**

Napisać program wyświetlający na ekranie tabelkę funkcji  $x \mapsto 3x^7 + (x - 4)^2 - x$  dla  $x \in \{0, \dots, 10\}$ .

*Wskazówka:* Wykorzystać schemat Hornera.

## **0x05 ZADANIE**

Napisać program, który czyta z wejścia liczby całkowite aż do napotkania liczby ujemnej a następnie zwraca największy/najmniejszy element z wczytanych liczb (z pominięciem ostatniej, ujemnej liczby).

## **0x06 ZADANIE**

Napisać program, który zamienia znaki tabulacji na odpowiednią liczbę znaków odstępu.

## **0x07 ZADANIE**

**Schemat Hornera**

Schemat Hornera jest sposobem obliczania wartości wielomianu zmniejszającym liczbę wykonywanych mnożeń, dzięki czemu skraca się czas działania programu. Ponadto w przypadku używania liczb zmiennoprzecinkowych, schemat Hornera ogranicza dodatkowo błędy zaokrągleń. Załóżmy, że mamy dany wielomian  $a_0 + a_1X + \dots + a_nX^n$ . Obliczenie wartości tego wielomianu w punkcie  $x$  schematem Hornera sprowadza się do wyliczenia:

$$a_0 + x \cdot (a_1 + x \cdot (a_2 + \dots x \cdot (a_{n-1} + x \cdot a_n) \dots))$$

Napisać program obliczający procentową częstotliwość występowania zadanego przez użytkownika znaku w linii tekstu.

**0x08 ZADANIE ★**

Napisać program, który odgaduje liczbę całkowitą z zakresu  $1, \dots, 100$  pomyślaną przez użytkownika. Użytkownik odpowiada jedynie czy zgadywana liczba jest za mała/za duża.

**0x09 ZADANIE**

Przeprowadzić analizę następującego programu

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     char a, x;
6     int l;
7
8     l = 0;
9     while( (x = getchar()) != '\n' ) {
10         if( l == 0 ) {
11             l = 1;
12             a = x;
13         } else {
14             l = 0;
15             putchar(x);
16             putchar(a);
17         }
18     }
19     return 0;
20 }
```