

## Tydzień 2; grupy podstawowe

03.03.2025

### Zadanie FuzzBizz

Napisz program, który dla liczb od 1 do  $n$  (gdzie  $n$  jest podane przez użytkownika) wypisuje:

- „**Fizz**”, jeśli liczba jest podzielna przez 3,
- „**Buzz**”, jeśli liczba jest podzielna przez 5,
- „**FizzBuzz**”, jeśli liczba jest podzielna zarówno przez 3, jak i przez 5,
- Samą liczbę, jeśli nie spełnia żadnego z powyższych warunków.

#### Przykładowe działanie programu:

Dla  $n = 15$ , program powinien wypisać:

```
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
```

## Zadanie Kwadraty

Napisz program szukający pierwiastków równania kwadratowego. Program powinien:

1. Wypisać komunikat:

```
Podaj współczynniki równania kwadratowego
Podaj a
```

Następnie wczytać wartość **a**. Analogicznie dla **b** i **c**.

2. Obliczyć wartość  $\Delta$  i wypisać ją na ekran.
3. Sprawdzić trzy przypadki:
  - $\Delta < 0 \rightarrow$  „Równanie nie ma rzeczywistych pierwiastków”.
  - $\Delta = 0 \rightarrow$  „Równanie ma jeden pierwiastek = ...”.
  - $\Delta > 0 \rightarrow$  „Równanie ma dwa pierwiastki:  $x_1 = \dots, x_2 = \dots$ ”.
4. Przetestować program dla danych:  $a = 1, b = -0.2, c = 0.01$ . Rozwiązanie powinno być jedno. Problem ten niwelujemy dodając epsilon np  $\epsilon = 10^{-9}$ .

## Zadanie Rok\_przestępca

Napisz program, który wczytuje ze standardowego wejścia rok, a następnie wypisuje na standardowe wyjście czy ten rok jest przestępny czy nie. Na początku powinno być sprawdzane, czy nie podaliśmy liczby mniejszej lub równej zero. W przypadku podania błędnych danych program powinien przerywać działanie z odpowiednim komunikatem (użyj polecenia `return`). Rok przestępny spełnia jeden z warunków:

- jest podzielny przez 4 z wyjątkiem lat podzielnych przez 100;
- jest podzielny przez 400.

## Zadanie Wyrazy

Napisz program, który ma za zadanie wczytać wyraz od użytkownika i wypisać go na ekranie. Użyj biblioteki `string`. Program powinien działać w nieskończonej pętli, aż użytkownik zdecyduje się zakończyć działanie programu. Program powinien:

1. Wczytać wyraz od użytkownika.
2. Jeśli użytkownik wpisze `exit`, program ma zakończyć działanie.
3. W przeciwnym razie program wypisuje na ekranie wprowadzony wyraz.

## Szczegóły implementacji

Program powinien działać w sposób ciągły, dopóki użytkownik nie zdecyduje się zakończyć go, wpisując słowo `exit`. W każdym innym przypadku program powinien wypisywać na ekranie wyraz, który został wprowadzony przez użytkownika.

## Przykład działania programu

```
Wpisz wyraz (lub 'exit' aby zakończyć): hello  
Wpisany wyraz: hello
```

```
Wpisz wyraz (lub 'exit' aby zakończyć): world  
Wpisany wyraz: world
```

```
Wpisz wyraz (lub 'exit' aby zakończyć): exit  
Koniec programu.
```

## Wymagania

- Program musi działać w pętli, wczytując dane z terminala.
- Program ma obsługiwać poprawnie sytuację, w której użytkownik wpisuje `exit`.
- Program ma wypisać wprowadzony przez użytkownika wyraz w każdej iteracji pętli, z wyjątkiem sytuacji, gdy użytkownik wpisze `exit`.