

# Laboratorium informatyki

## Programowanie w języku C

### Ćwiczenie 4

#### 1. Decyzja

Jedną z najistotniejszych cech języków programowania jest możliwość podejmowania przez algorytm decyzji w zależności od spełnionych kryteriów wejściowych. Dzięki temu program potrafi reagować na decyzje użytkownika bądź parametrów zewnętrznych (stanu portów itp.). Podstawową konstrukcją decyzyjną w języku C jest instrukcja warunkowa `if`.

#### 2. Instrukcja warunkowa

Instrukcja warunkowa `if` wykonuje wybrany fragment kodu gdy warunek w niej opisany zwróci wartość `true`. Warunek może być opisany relacją lub funkcją. W instrukcji `if` można zapisać podstawowe warunki relacyjne:

`==` - relacja równości  
`<` - mniejsze niż  
`>` - większe niż  
`<=` - mniejsze lub równe  
`>=` - większe lub równe  
`!=` - różne od

Wartości zwracane przez funkcje także mogą być sprawdzane warunkiem relacji. Funkcje typu `bool` nie muszą podlegać porównaniu relacyjnemu. Instrukcja warunkowa `if` może sprawdzać wartość jaką zwraca taka funkcja i odpowiednio reagować. Możliwe jest zaprogramowanie reakcji na wartość `true` lub `false` wstawiając do instrukcji warunkowej negację funkcji.

Możliwe jest budowanie złożonych warunków za pomocą operatorów logicznych `&&` - logiczne „i”, `||` - logiczne „lub”.

Podstawowa składnia instrukcji warunkowej:

```
if (warunek) kod_wykonywany_gdy_warunek_ true;
```

Przy spełnionym warunku może być wykonywana jedna instrukcja kodu lub cały blok kodu ograniczony `{}`. Możliwe jest rozbudowanie instrukcji warunkowej o fragment realizujący określony kod gdy warunek zwróci wartość `false`. Realizowane to jest za pomocą instrukcji `else`:

```
if (warunek) {  
    kod_gdy true;  
}else{  
    kod_gdy false;  
}
```

Z wykorzystaniem instrukcji `else` można budować kaskadę warunków:

```
if(warunek1) {  
    kod gdy true;  
}else if(warunek2) { //gdy warunek1 zwróci false  
    kod gdy warunek2 zwróci true;  
}else{  
    kod gdy warunek2 zwróci false;  
}
```

Taka kaskada może analizować praktycznie dowolną liczbę powiązanych ze sobą warunków.

### 3. Zadania

Wykorzystując zagadnienia omówionych w materiałach wstępnych do niniejszej instrukcji napisać programy realizujące poniższe zadania. Wydzielić dla pisanego programu plik nagłówkowy, poszczególne zadania realizować z wykorzystaniem funkcji:

- a) Napisać program wyznaczający pierwiastki równania kwadratowego.
- b) Napisać program analizujący trzy podane przez użytkownika liczby i wyświetlający je w kolejności rosnącej.