

# Laboratorium 2

---

## 1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie się z obsługą C. Procedury i funkcje. Pętle **for**, instrukcja warunkowa **if**.

Przykładowo, nie można wykonać operacji kopiowania czy przesunięcia pliku do innego katalogu, jeśli nie jest znana nazwa tegoż pliku oraz nazwa docelowego folderu. Gdybyśmy napisali funkcję realizującą taką czynność, to nazwy pliku oraz katalogu finalnego byłyby jej **parametrami**.

**Parametry funkcji** to dodatkowe dane, przekazywane do funkcji podczas jej wywołania.

Parametry pełnią rolę dodatkowych zmiennych wewnątrz funkcji i można ich używać podobnie jak innych zmiennych, zadeklarowanych w niej bezpośrednio.

### (Przykład 1 Język C++)

```
#include <iostream>
void Dodaj(int nWartosc1, int nWartosc2)
{
    int nWynik = nWartosc1 + nWartosc2;
    std::cout << nWartosc1 << " + " << nWartosc2 << " = " << nWynik;
    std::cout << std::endl;
}
int main()
{
    int nLiczba1;
    std::cout << "Podaj pierwsza liczbe: ";
    std::cin >> nLiczba1;
    int nLiczba2;
    std::cout << "Podaj druga liczbe: ";
    std::cin >> nLiczba2;
    Dodaj(nLiczba1, nLiczba2);
}
```

### Wartość zwracana przez funkcję

Spora część funkcji pisanych przez programistów ma za zadanie obliczenie jakiegoś wyniku (często na podstawie przekazanych im parametrów). Inne z kolei wykonują operacje, które nie zawsze muszą się udać (choćby usunięcie pliku – dany plik może przecież już nie istnieć). W takich przypadkach istnieje więc potrzeba, by funkcja zwróciła jakąś wartość. Niekiedy będzie to rezultat jej intensywnej pracy, a innym razem jedynie informacja, czy zlecona funkcji czynność została wykonana pomyślnie. Zgodnie ze zwyczajem, popatrzymy teraz na odpowiedni program przykładowy:

**(Przykład 2 Język C++)**

```
#include <iostream>
```

```
int Obwod(int nBok1, int nBok2)
{
    return 2 * (nBok1 + nBok2);
}
int Pole(int nBok1, int nBok2)
{
    return nBok1 * nBok2;
}
int main()
{
    int nDlugosc1;
    std::cout << "Podaj dlugosc pierwszego boku: ";
    std::cin >> nDlugosc1;
    int nDlugosc2;
    std::cout << "Podaj dlugosc drugiego boku: ";
    std::cin >> nDlugosc2;
    std::cout << "Obwod prostokata: " << Obwod(nDlugosc1, nDlugosc2) <<
    std::endl;
    std::cout << "Pole prostokata: " << Pole(nDlugosc1, nDlugosc2) <<
    std::endl;
    return 0;
}
```

Instrukcja warunkowa **if, else.**

**(Przykład 3 Język C++)**

```
#include <iostream>
```

```
int main()
{
    int nLiczba;
    std::cout << "Wprowadz liczbe wieksza od 10: ";
    std::cin >> nLiczba;
    if (nLiczba > 10)
    {
        std::cout << "Dziekuje." << std::endl;
        std::cout << "Wcisnij dowolny klawisz, by zakonczyc.";
    }
    else {
        std::cout << "Liczba " << nLiczba
        << " nie jest wieksza od 10." << std::endl;
        std::cout << "Czuj sie upomniany :P";
    }
}
```

## 2. Przykłady - Język C

### Przykład4

Napisać program, wypisujący liczby od 1 do 100 (Użyć pętli „for”).

```
#include <stdio.h>
int main(void){
    int i;
    for(i=1;i<101;i=i+1)printf("\n %d",i);
    return 0;
}
```

### Przykład5

Napisać program, wypisujący liczby między 17 a 100, które są podzielne przez 17 (Użyć pętli „for”).

```
#include <stdio.h>
int main(void){
    int i;
    for(i=17;i<101;i=i+1)
        if((i%17)==0)printf("\n %d",i);
    return 0;
}
```

### Przykład6

Znajdywanie minimum z 3 liczb.

```
#include<stdio.h>
int main()
{
    float a, b, c, min;
    printf("Podaj 3 liczby:\n");
    printf("a = ");
    scanf("%f", &a);
    printf("b = ");
    scanf("%f", &b);
    printf("c = ");
    scanf("%f", &c);
    if(a < b)
    if(a < c) min = a;
    else min = c;
    else
    if(b < c) min = b;
    else min = c;
    printf("Minimum: %.2f\n", min);
    return 0;
}
```

**Przykład7**

Napisać program konwertujący liczbę wprowadzoną przez użytkownika w postaci szesnastkowej do odpowiedniej postaci ósemkowej i dziesiętnej (zastosować funkcję printf() ze specyfikatorami

%u -liczba całkowita bez znaku w postaci dziesiętnej,  
%x -liczba całkowita bez znaku w postaci szesnastkowej,  
%o – liczba całkowita bez znaku w postaci ósemkowej)..

```
#include <stdio.h>
int main(void){
    unsigned u;
    printf("Wprowadz liczbe szesnastkowo:\n");
    scanf("%x",&u);
    printf("Odpowiednik dziesiętny: %u",u);
    printf("\nOdpowiednik osemkowy: %o",u);
    return 0;
}
```

**Przykład8**

Napisać program wymagający podania personaliów, a następnie wczytujący imię, drugie imię i nazwisko. Program powinien wypisać wprowadzoną informację.

**Rozw1 (Język C)**

```
#include <stdio.h>

int main(void){
    char imie[21], drugie[21], nazwisko[21];
    printf("Podaj pelne personalia(imie, drugie imie, nazwisko):\n");
    gets(imie);
    gets(drugie);
    gets(nazwisko);
    printf("%s %s %s", imie, drugie, nazwisko);
    return 0;
}
```

**Rozw2 (Język C)**

```
#include <stdio.h>
int main(void){
    char imie[21], drugie[21], nazwisko[21];
    printf("Podaj pelne personalia(imie, drugie imie, nazwisko):\n");
    scanf("%20s%20s%20s", imie,drugie, nazwisko);
    printf("%s %s %s", imie, drugie, nazwisko);
    return 0;
}
```

//Zamiast %20s można użyć %s

**Rozw3 (Język C++)**

```
#include <iostream>
using namespace std;
int main(void){
    char imie[21], drugie[21], nazwisko[21];
    cout<<"Podaj pelne personalia(imie, drugie imie, nazwisko):\n";
    cin>> imie >> drugie >> nazwisko;
    cout<<imie <<" " << drugie <<" " << nazwisko;
    return 0;
}
```

**3. Program ćwiczenia****Zad1**

Napisać program, korzystający z funkcji o nazwie **przeciw()**, zwracającej długość przeciwprostokątnej w trójkącie prostokątnym po dodaniu długości dwóch przyprostokątnych. Funkcja powinna zwracać wartość typu **double**. Argumenty funkcji powinny być typu **double** np. `double przeciw(double s1, double s2)`. Należy zademonstrować działanie twierdzenia Pitagorasa.

**Zad2**

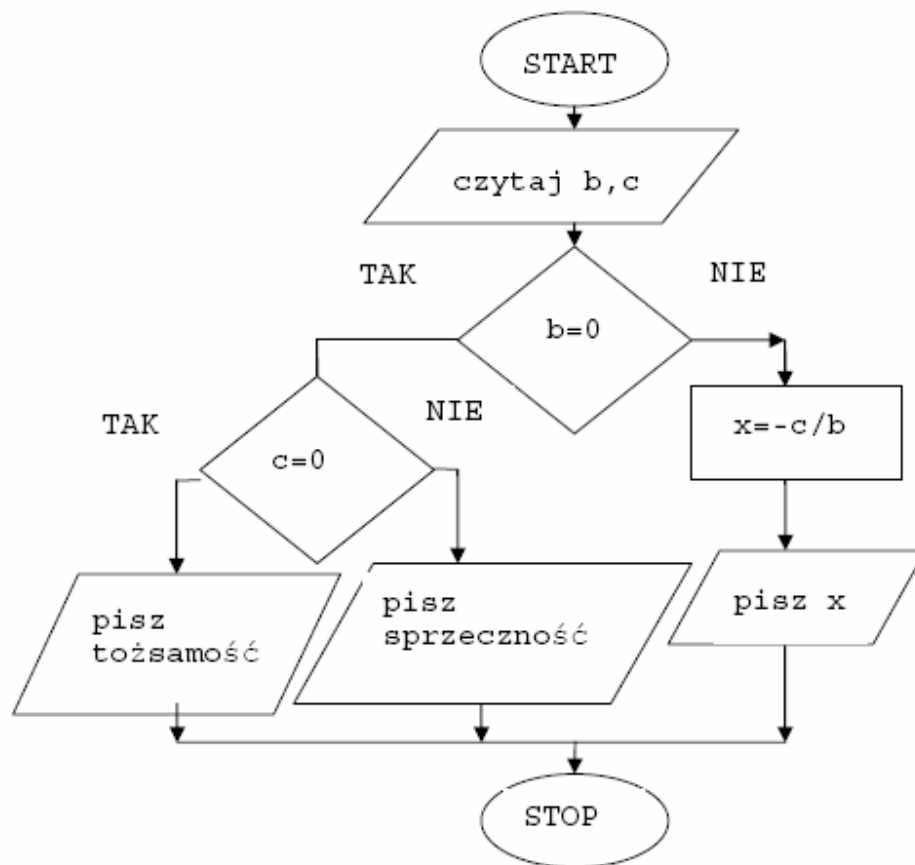
Napisać program konwertujący liczbę wprowadzoną przez użytkownika w postaci ósemkowej do odpowiedniej postaci szesnastkowej i dziesiętnej.

**Zad3**

Napisać program wymagający podania imion babci i dziadka. Program powinien wypisać wprowadzoną informację.

**Zad4**

Napisać program do liczenia pierwiastka równania  $bx+c=0$ .

**Zad5**

Jednym z możliwych zastosowań instrukcji warunkowej jest sprawdzanie, czy zadany punkt płaszczyzny mieści się określonym obszarze ograniczonym przez wykresy funkcji. Przykładowy program wczytuje dwie liczby rzeczywiste, będące współrzędnymi  $x$  i  $y$  punktu na płaszczyźnie i sprawdza, czy ten punkt należy do obszaru oznaczonego na poniższym rysunku kolorem żółtym. Obszar jest ograniczony przez wykresy trzech funkcji:

$$y = 10x$$

$$y = x^3 + 3$$

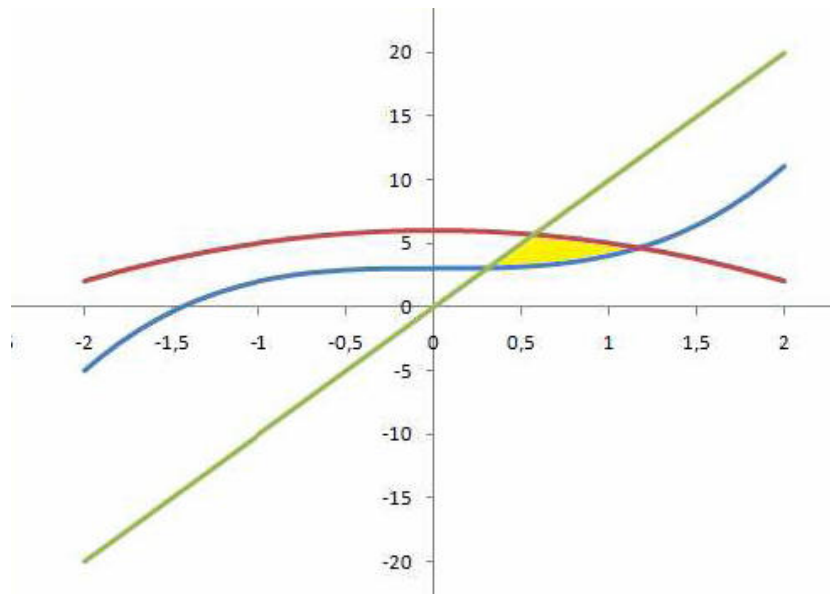
$$y = -x^2 + 6$$

Jak wynika z rysunku, należy sprawdzić, czy zadany punkt leży pod wykresem funkcji kwadratowej:  $y = -x^2 + 6$ , nad wykresem funkcji  $y = x^3 + 3$  oraz pod wykresem liniowej funkcji:  $y = 10x$ , czyli, czy współrzędne punktu  $x$  i  $y$  spełniają następujące nierówności:

$$y < 10 * x$$

$$y < -x * x + 6$$

$$y > x * x * x + 3$$



Np.  $x=0.5$ ,  $y=4$

### Zad6

Napisać program z pętlą „for”, który oblicza sumę  $n$  liczb wprowadzonych z klawiatury.

Dane:  $n$  liczb

Wczytaj  $n$  liczb i policz ich sumę

