

# Laboratorium informatyki

## Programowanie w języku C

### Ćwiczenie 6

#### 1. Pętle

Opracowując algorytm rozwiązania danego problemu często przewiduje się konieczność kilkukrotnego wykonania określonego kodu. W takich sytuacjach pomocne są konstrukcje pętli. W poprzednim ćwiczeniu omówiono wczesną formę konstrukcji pętli realizowaną za pomocą połączenia instrukcji warunkowej i instrukcji skoku. W języku C i pokrewnych dostępne są jeszcze dedykowane instrukcje pętli: `for()` – pętla iteracyjna i dwie pętle warunkowe `while()` i `do while()`.

#### 2. Pętla iteracyjna

Pętla iteracyjna `for()` umożliwia zaprogramowanie wykonania przypisanego bloku kodu określoną liczbę razy. Zazwyczaj pętla `for()` jest realizuje swoje zadanie w oparciu o wewnętrzną zmienną sterującą. Z każdym powtórzeniem pętli zmienna ta jest inkrementowana lub dekrementowana. Iteracje pętli trwają tak długo jak spełniony jest warunek wywołania kolejnej inkrementacji.

`for(start; koniec; inkrementacja) {}`

`start` – inicjacja zmiennej sterującej

`koniec` – warunek trwania pętli

`inkrementacja` – algorytm inkrementacji lub dekrementacji zmiennej sterującej

```
C petlaFOR.c > ...
1  #include <stdlib.h>
2  #include <stdio.h>
3  int main()
4  {
5      for(int i=1;i<=5;i++)
6          printf("Wykonano %i iterację pętli for.\n",i);
7  }
```

Wykonano 1 iterację pętli for.  
Wykonano 2 iterację pętli for.  
Wykonano 3 iterację pętli for.  
Wykonano 4 iterację pętli for.  
Wykonano 5 iterację pętli for.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3  int main()
4  {;
5      for(int i=1,j=1;j<=20;i++,j+=i)
6          printf("Wykonano %i iterację pętli for dla zmiennych i=%i i j=%i.\n",i,i,j);
7  }
```

petlaFOR

Wykonano 1 iterację pętli for dla zmiennych i=1 i j=1.  
Wykonano 2 iterację pętli for dla zmiennych i=2 i j=3.  
Wykonano 3 iterację pętli for dla zmiennych i=3 i j=6.  
Wykonano 4 iterację pętli for dla zmiennych i=4 i j=10.  
Wykonano 5 iterację pętli for dla zmiennych i=5 i j=15.

W pewnych sytuacjach pomimo zaprogramowania określonej liczby iteracji może pojawić się konieczność przerwania wykonywania pętli. Język C dostarcza kilku narzędzi do realizacji tego zadania.

`break;` - instrukcja wyjścia z bloku kodu, przerywa działania pętli, realizowany jest kod zapisany po pętli.

`continue;` - przerwanie aktualnej iteracji pętli.

`return wartość;` - wyjście z programu.

```
1  #include <stdlib.h>
2  #include <stdio.h>
3  int main()
4  {
5      for(int i=1;i<20;i++){
6          if(i%2==1) continue;
7          if(i==18) break;
8          printf("Wykonano %i iterację pętli for.\n",i);
9      }
10 }
```

Wykonano 2 iterację pętli for.  
Wykonano 4 iterację pętli for.  
Wykonano 6 iterację pętli for.  
Wykonano 8 iterację pętli for.  
Wykonano 10 iterację pętli for.  
Wykonano 12 iterację pętli for.  
Wykonano 14 iterację pętli for.  
Wykonano 16 iterację pętli for.

### 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:

- Napisać program obliczający silnię podanej liczby
- Napisać program obliczający średnią arytmetyczną liczb parzystych i średnią geometryczną liczb nieparzystych z zadanego przedziału