

Przykład1:

Napisz funkcję, która obliczy silnię dowolnej liczby naturalnej.

1. Silnia obliczana iteracyjnie:

Silnią dodatniej liczby naturalnej nazywamy iloczyn kolejnych liczb naturalnych, począwszy od 1 aż do liczby, dla której obliczamy wartość silni .

Formalny zapis matematyczny funkcji *silnia* może wyglądać następująco:

$$n! = 1$$

$$n! = 1 * \dots * n$$

```
double silnia_iteracyjnie(int n)
{
    double silnia = 1;
    for (int i=2; i<n+1; i++)
        silnia = silnia *i;

    return silnia;
}
```

Rekurencja

W informatyce funkcją rekurencyjną jest funkcja, która wywołuje samą siebie.

Przykład2:

Silnia obliczana rekurencyjnie

W ogólnej postaci można to zapisać, jako: $n! = (n-1)! * n$

Aby funkcja rekurencyjna była poprawna, należy ściśle zdefiniować warunek zakończenia wywołań rekurencyjnych.

Pełna definicja rekurencyjna funkcji *silnia* ma postać:

$n! = 1$

$n! = (n-1)! * n$

```
1  double silnia(int n)
2  {
3      if (n==0)          // warunek początkowy rekurencji
4          return 1;
5      else
6          return silnia (n-1) *n; // rekurencyjne wywołanie funkcji
7  }
8
```

W linii 6 zawarte jest wywołanie rekurencyjne funkcji *silnia*. We wnętrzu funkcji wywołujemy funkcję, którą właśnie piszemy. Taki zapis powoduje, że program po dotarciu do linii numer 6 kodu nie jest w stanie dokończyć instrukcji, musi bowiem wywołać funkcję *silnia* z parametrem n o 1 mniejszym. Każde wywołanie funkcji jest związane z zapamiętaniem kopii wszystkich zmiennych funkcji oraz z umieszczeniem na stosie adresu powrotnego, czyli miejsca w programie, do którego system ma powrócić po zakończeniu funkcji.

Przykład3:

Napisz program obliczający n-ty wyraz ciągu Fibonacciego:

- metodą iteracyjną;
- metodą rekurencyjną.

```
7 int fibonacci (int n)
8 {
9     if(n<2)
10         return 1;
11     else
12         return fibonacci(n-1)+fibonacci(n-2);
13 }
```

Przykład4:

```
1 #include <iostream>
2 #include <cstdio>
3 using namespace std;
4
5 void odliczanie (int n)
6 {
7     cout<<"ODLICZAM....."<<n<<endl;
8     if (n>0)
9         odliczanie (n-1);
10    cout<<n<<" - W maju matura "<<endl;
11 }
12 int main()
13 {
14     odliczanie(4);
15
16
17     cin.ignore();
18     getchar();
19     return 0;
20 }
21
```

Zadania:

1. Napisz funkcję obliczającą dla liczby rzeczywistej wartość potęgi o wykładniku naturalnym.

$$a^0 = 1$$

$$a^n = a^{n-1} * a$$

Funkcja pobiera 2 argumenty

2. Napisz program, który zamieni liczbę z postaci dziesiętkowej na liczbę o innej podstawie liczenia.