

Programownie I

Wykład 3

dr inż. Adam Zielonka

Instytut Matematyki,
Politechnika Śląska

Gliwice 19.10.2018

Funkcje

Deklaracja funkcji

```
typ_zwracany nazwaFunkcji(typ_arg1,typ_arg2);  
typ_zwracany nazwaFunkcji(typ_arg nazwa_arg);  
void nazwaFunkcji(void);  
void nazwaFunkcji();
```

przykład

```
double suma(double, double);  
double suma(double x, double y);  
void drukujRamke(void);  
void drukujRamke();
```

Funkcje

Definicja funkcji

```
typ_zwracany nazwaFunkcji( typ_arg1 nazwa1 , typ_arg2 nazwa2 )  
{  
    typ_zwracany zmienna;  
    //instrukcje  
    return zmienna;  
}
```

Przykład

```
double suma(double x, double y)  
{  
    return x+y;  
}
```

Funkcje

Przykład

```
#include<iostream>
using namespace std;
double suma(double, double); //deklaracja funkcji

int main()
{
    double x{},y{},z{};
    cin >> x >> y;
    z=suma(x,y); //wywołanie funkcji
    cout << x <<'+'<< y<< '=' << z <<endl;
}
//definicja funkcji
double suma(double x, double y)
{
    return x+y;
}
```

Przesyłanie argumentów przez referencję

Przykład

```
#include<iostream>
using namespace std;
void napis(int&);
int main(){
    int ilosc=0,i=0;
    for( ; i<20 ; i++)
    {
        cout << napis(ilosc) <<endl;
        cout<< ilosc << endl;
    }
    system("PAUSE");
}
void napis(int &licznik) // &licznik jest adresem do zmiennej w argumencie
{
    licznik++;
    cout<<"Witam"<<endl;
}
```

Zmienne lokalne

Przykład

```
#include<iostream>
using namespace std;
long long silnia( unsigned int);
int main(){
    int i=0; //zmienna lokalna funkcji main
    for( ; i<20 ; i++)
        cout << silnia(i) <<endl;
    system("PAUSE");
}
long long silnia(unsigned int n) // n - zmienna lokalna
{
    n++;
    long long wynik=1; //zmienna lokalna
    for( int i=2 ; i<n; i++ )
        wynik*=i;
    return wynik;
}
```

Zmienne globalne

Przykład

```
#include<iostream>
using namespace std;
int a=0;
void drukuj( int );
int main(){
    drukuj(0);          // drukuje 0
    a++;
    drukuj(1);          // drukuje 2
    drukuj(100);       // drukuje 102
    system("PAUSE");
}
void drukuj(int n)
{
    a+=n;
    cout<< a<<endl;
}
```

Zmienne statyczne

Przykład

```
#include<iostream>
using namespace std;
void drukuj();
int main(){
    drukuj();          // drukuje 1
    for(int i=0; i<10; i++)
        cout << drukuj() <<endl;
    system("PAUSE");
}
void drukuj()
{
    static int iloscWywolan=0;
    cout << ++iloscWywolan << endl;
}
```


Argumenty domniemane

Schemat

```
typF nazwaFunkcji( typ1 = wartos1, typ2 = wartosc2);  
...  
typF nazwaFunkcji( typ1 nazwa1, typ2 nazwa2)  
{  
    ...  
    return wyn;  
}
```

Uwaga! Określenie wartości domyślnej w deklaracji i definicji tego samego argumentu spowoduje błąd.

```
int suma(int, int=1);  
...  
int suma(int x, int y=1) //redefinition of default argument  
{ return x + y; }
```

Argumenty domniemane

Przykład

```
#include<iostream>
using namespace std;
double suma(double x=0,double y=0,double z=0);
int main()
{ double x=1.0,y=2.0,z=3.0;
  cout << suma() <<" " << suma(x) << " ";
  cout << suma(x,y) << " " << suma(x,y,z);
  system("PAUSE");
}
double suma(double x,double y,double z)
{
  return x+y+z;
}
```

Tablice

Schemat

```
typ nazwa[ilosc_elementow];  
typ nazwa[] = {wartosc1, wartosc2, ..., wartoscN};  
typ nazwa[n] = {wartosc1};
```

Przykład

```
double imie[10];  
//tablica 10 zmiennych typu double  
//wypełniona przypadkowymi wartościami  
  
int elementy[] = {2, 3, 1, 3, 0};  
  
int elementy[5] = {1,4}; //{1,4,0,0,0} dopełnia zerami
```

Tablice

Ogólnie informacje

- Tablice indeksowane są od 0 do $n - 1$.
- do i -tego wyrazu tablicy odwołujemy się nazwa[i]
- kompilator nie czuwa nad tym, czy przekroczyliśmy zakres indeksu

```
int a[4]={1};  
int b=a[4]+a[12];
```

Tablice znakowe

```
char imie[20];  
//tablica 20 zmiennych typu char  
  
char nazwisko[]="Zielonka";  
//po ostatnim znaku wstawiony zostanie znak '\0'  
// o wartości liczbowej 0
```

Z	'	i	'	e	'	l	'	o	'	n	'	k	'	a	'	\0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----

90	105	101	108	111	110	107	97	0
----	-----	-----	-----	-----	-----	-----	----	---

Łańcuchy znaków

Przykład

```
char nazwaMiasta[30];
cout<<"Podaj nazwe miasta: ";
cin.getline(nazwaMiasta,30);

//wypisanie nazwy na ekran
cout<<nazwaMiasta<<endl;

//wypisanie po jednym znaku
int i=0;
while(nazwaMiasta[i]!=0)
    cout<<nazwaMiasta[i++];
cout<<endl;
```

Uwaga

```
char wyraz1[] = "Ala";  
char wyraz2[] = "Ada";  
if(wyraz1==wyraz2) polecenie; \\ŹŁE
```

Przykład

```
bool czyRowne(char[],char[]);  
...  
bool czyRowne(char w1[] ,char w2[] )  
{  
    int i =0;  
    while( w1[i] != 0 && w2[i] != 0 )  
        if( w1[i] != w2[i++] ) return false;  
    if( w1[i] == 0 && w2[i] == 0 ) return true;  
    return false;  
}
```

cstring

```
#include<cstring>
//...
char wyraz1[] = "Ala";
char wyraz2[] = "Ada";

if(strcmp(wyraz1 ,wyraz2)==0)
    cout <<"takie same wyrazy";
```


Tablice wielowymiarowe

Schemat

```
typ nazwa[ n ][ m ];  
nazwa[ i ][ j ]=wartosc;
```

Przykłady

```
int liczby[2][3]={{1,2,3},{1,1,1}};
```

```
double x[3][2]={{1.0,2.0}};
```

```
double y[2][3]={{1.0},{1.0}};
```

```
int [2][3]={1,2,3,4,5,6};
```

Przykład

```
const int n=10, m=10;
int tabliczka[n][m];

for( int i=0 ; i<n ; i++)
{
    for(int j=0 ; j<m ;j++)
    {
        tabliczka[i][j] = ( i<=j ? i*j : 0);
        cout.width(3);
        cout << right << tabliczka[i][j] <<' ';
    }
    cout << endl;
}
```

Przykład

```
char dane [2] [10]={"Adam","Zielonka"};
```

'A'	'd'	'a'	'm'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
'Z'	'i'	'e'	'l'	'o'	'n'	'k'	'a'	'\0'	'\0'