



Podstawy programowania

Środowiska projektowe IDE. Organizacja projektu.

Zmienne i ich typy. Operacje wejścia/wyjścia w C++



mgr inż. Marta Okoń-Fąfara

mail: marta.okon@wat.edu.pl

tel: 261 83 92 43

pok: 121/61

Konsultacje: wtorek 15.00-16.30

czwartek 9.50-11.40

mgr inż. Michał Knioła

mail: michal.kniola@wat.edu.pl

tel: 261 83 92 38

pok: 44/54

Konsultacje: środa 13.30-15.30

czwartek 11.40-13.15



Tematy ćwiczeń laboratoryjnych:

1. Środowiska projektowe IDE. Organizacja projektu. /2
2. Zmienne i ich typy. Operacje wejścia/wyjścia w C++ /2
3. Analiza algorytmów. Projektowanie algorytmów. /4
4. Sterowanie w programie i śledzenie przebiegu obliczeń /4
5. Instrukcje obliczeń cyklicznych w C++. Zastosowanie pętli for oraz tablic /4
6. Instrukcje obliczeń cyklicznych w C++. Zastosowanie pętli while oraz pętli do while w programach /4
7. Kolokwium 1
8. Projektowanie wybranych aplikacji w języku C++ /4
9. Funkcje w C++. Projektowanie aplikacji /4
10. Funkcje w C++. Zastosowanie wskaźników przy przekazywaniu danych /4
11. Elementy programowania obiektowego w C++ /4
12. Projektowanie programu z komponentami VCL /2
13. Kolokwium zaliczające laboratorium /2



Literatura:

1. Walczak-Struzińska Anna, Walczak Krzysztof, Nauka programowania dla początkujących C++, Wydawnictwo W&W ion 2016
2. Stasiewicz Andrzej, C++ Ćwiczenia 2004, Helion 2004
3. Grębosz Jerzy, Symfonia C++ Standard. Programowanie w języku C++ orientowane obiektowo, Edition 2000

Warunki zaliczenia

- Kartkówki
- Ocena za zadanie wykonane podczas zajęć (informacja na początku zajęć)
- Kolokwium (zadania do wykonania – na kartce)
- Wszystkie oceny muszą być pozytywne
- Ocena końcowa to średnia arytmetyczna ocen cząstkowych



Środowiska programistyczne

- Dev C++
- Embarcadero RAD Studio
- Eclipse (portable)
- Code Blocks
- Qt Designer
- Visual Studio
- NetBeans



Zad.1. Utworzyć folder na pulpicie (nazwa: imię i nazwisko studenta), wewnątrz utworzyć podfolder o nazwie *HelloWorld*. Napisać program wyświetlający na ekranie napis *Hello World!*

Zad.2. We własnym folderze utwórz podfolder na program, który:

- Zadeklaruje zmienne następujących typów:
 - Binarną b
 - Znakową c
 - Całkowitoliczbową i
 - Całkowitoliczbową długą li
 - Zmiennoprzecinkową f
 - Zmiennoprzecinkową podwójnej precyzji d
 - Zmiennoprzecinkową podwójnej precyzji długą ld
- Wyświetli zajętość pamięci każdej ze zmiennych.



```
1  #include <iostream.>
2  //using namespace std;
3  int main(int argc, char** argv) {
4  // ponizej deklaracje roznych typow danych
5  bool b;
6  char c;
7  int i;
8  long int li;
9  float f;
10 double d;
11 long double ld;
12 // Teraz wydruki zajetosci pamieci
13 std::cout << " ILE BAJTOW ZAJMUJA ZMIENNE\n";
14 std::cout << " W PAMIECI KOMPUTERA W C++:\n\n";
15 std::cout << "bool - " << sizeof(b) << std::endl;
16 std::cout << "char - " << sizeof(c) << std::endl;
17 std::cout << "int - " << sizeof(i) << std::endl;
18 std::cout << "long int - " << sizeof(li) << std::endl;
19 std::cout << "float - " << sizeof(f) << std::endl;
20 std::cout << "double - " << sizeof(d) << std::endl;
21 std::cout << "long double- " << sizeof(ld) << std::endl;
22 //system("pause");
23 return 0;
24 }
```



Typ	Znak	Ilość bajtów	Zakres wartości
char	jest	1	0... + 255
int	jest	2	-32768...+32767
short			jw
long	jest	4	-2 147 483 648 ... 2 147 483 647
unsigned char	-	1	0-255
unsigned int	-	2	0- 65 535
unsigned short	-	2	jw
enum	-	2	jw
float	jest	4	-3.4E+38 + 3.4E+38 (dokładność: 7 cyfr)
double	jest	8	-1.7E+308... + 1.7E+308 (dokładność: 15 cyfr);
long double	jest	10	-3.4E+4932 ...+3.4E+4932
void	-		dotyczy parametrów funkcji



Zad.3. We własnym folderze utwórz podfolder na program, który:

- Zadeklaruje liczbę zmiennoprzecinkową pojedynczej precyzji x
- Zadeklaruje dwie liczby typu całkowitoliczbowego $i1$, $i2$
- Pobierze wartość każdej zmiennej z klawiatury

(przykładowy komunikat: *Wprowadź wartość liczby zmiennoprzecinkowej: itp.*)

- Wyświetli wartość każdej zmiennej

(przykładowy komunikat: *Wprowadzona wartość liczby zmiennoprzecinkowej: itp.*)

Wykorzystaj funkcje:

`printf()`

`scanf()`



```
U_scanf1.cpp
//  WCZYTYWANIE LICZB
//  FUNKCJE: scanf printf

#include <stdio.h>
#include <conio.h>

main()
{
    int I1, I2;
    float X;
    printf(" WPROWADZANIE LICZB FUNKCJA scanf()");
    printf("\n\n DALEJ WPISUJ    LICZBY CALKOWITE!\n\n");

    printf("wpisz I1: ");
    scanf("%d",&I1);
    printf("wprowadzono      I1= %d \n",I1);

    printf("\n\nwpisz I1 spacja I2: ");
    scanf("%d %d", &I1, &I2);
    printf("wprowadzono: I1= %d I2= %d ",I1,I2);

    printf("\n\n WPISZ    LICZBE RZECZYWISTA:\n\n");
    printf("wpisz X: ");
    scanf("%f",&X);
    printf("wprowadzono      X= %5.3f \n",X);

    getch();
}
```

```
C:\Program Files\Borlan...
WPROWADZANIE LICZB FUNKCJA scanf()
DALEJ WPISUJ    LICZBY CALKOWITE!

Wpisz I1: 34
wprowadzono      I1= 34

Wpisz I1 spacja I2: 45 67
wprowadzono: I1= 45 I2= 67

WPISZ    LICZBE RZECZYWISTA:

Wpisz X: 123.456
wprowadzono      X= 123.456
```



Zad 4.

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int a,b;
5      cout<< "\a \n PRZYKLAD 1 \n"; //a pisk!
6      a = 10;
7      b = a++; // operator a++ przyrostkowy
8      cout << "\nTu a= "<< a <<" b= " <<b<<"\n";
9      a = 10;
10     b = ++a; // operator ++a przedrostkowy
11     cout << "\nTu a= "<< a <<" b= " <<b<< endl;
12     cout<< "\a \n PRZYKLAD 2 \n"; //a pisk!
13     a=10;
14     b=a--;
15     cout << "\nTu a= "<< a <<" b= " <<b<< endl;
16     a=10;
17     b=--a;
18     cout << "\nTu a= "<< a <<" b= " <<b<< endl;
19     cout<< "\n PRZYKLAD 3 \n";
20     a = 10;
21     b = (a++, a+=10); // operator a++
22     cout << "\na= "<< a <<" b= " <<b <<endl;
23     a=10;
24     b = (a++ + (a+=10));
25     cout << "\na= "<< a <<" b= " <<b <<endl;
26     a = 10;
27     b = (++a, a+=10); // operator ++a
28     cout << "\na= "<< a <<" b= " <<b <<endl;
29     return 0;
30 }
```