**Міністерство освіти і науки, молоді та спорту України**

**Національний університет „Львівська політехніка”**

**Кафедра ЕОМ**

****

**Звіт**

**з лабораторної роботи №1**

**з дисципліни: “Програмування (ООП)”**

**на тему: “ЛАБОРАТОРНА РОБОТА №1. ПЕРЕВАНТАЖЕННЯ ФУНКЦІЙ”**

**Варіант 22**

**Виконав: ст.гр. КІ-15**

**Петрущак Я.В.**

**Прийняв: викладач**

**Козак Н.Б.**

**Львів**

**2020**

**Теоретичні відомості**

**Перевантаження функцій**

У мові С++, на відміну від мови С дозволяється визначати декілька функцій з одним і тим же іменем за умови, що дані функції мають різну сигнатуру (різні типи та кількість аргументів функції). При цьому тип значення що повертається при перевантаженні до уваги не приймається.

У мові С директива препроцесора #defіne використовується для визначення констант та макросів (макровизначень). Наприклад, директива 7 #defіne sqr(x) ((x)\*(x)) оголошує макрос, що дозволяє обчислювати квадрат від фактичного аргументу, який підставляється замість формального аргументу х

Мова С++ пропонує безпечну заміну макросам – вбудовані (іnlіne) функції. На відміну від макросів, вбудовані функції не піддаються помилкам подвійного обчислення. Крім того типи аргументів перевіряються компілятором і при потребі виконуються всі необхідні перетворення. Таким чином, якщо у вас є маленька функція (кілька рядків коду), яку доводиться часто викликати, то її можна оголосити як іnlіne. Оголошена в такий спосіб функція буде проаналізована компілятором на можливість її реалізації у вигляді вбудованої функції. Якщо компілятор вважатиме доцільним реалізувати її як вбудовану, то дана функція не буде викликатися. Замість цього тіло функції підставлятиметься в те місце програми, де здійснюється виклик. При цьому підвищується ефективність програми ціною збільшення розміру коду програми.

**Варіант 22 – код програми.**

[#include](tg://search_hashtag?hashtag=include) <stdio.h>  
[#include](tg://search_hashtag?hashtag=include) <conio.h>  
[#include](tg://search_hashtag?hashtag=include) <string.h>  
[#include](tg://search_hashtag?hashtag=include) <clocale>  
  
using namespace std;  
  
  
[#define](tg://search_hashtag?hashtag=define) N 10  
  
int res = 0;  
  
int func(double\* mas, int B)

{  
int high = 0;  
int low = mas[B - 1];  
 for (int i = 0; i < B; i++)

{  
 if (mas[i] > mas[high]) high = i;  
 if (mas[i] < mas[low]) low = i;  
 }  
 if (low > high)

{  
 int a = low;  
 low = high;  
 high = a;  
 }  
 for (int i = low + 1; i < high; i++)

{  
 if (mas[i - 1] < mas[i] && mas[i + 1] < mas[i]) res++;  
 }  
return res;  
}  
  
int func(char\* str)

{  
 int b = (sizeof(str) / sizeof(char)) - 2;  
 int high = 0;  
 int low = str[b - 1];  
 for (int i = 0; i < b; i++)

{   
 if (str[i] > str[high]) high = i;  
 if (str[i] < str[low]) low = i;  
 }  
 if (low > high)

{  
 int a = low;  
 low = high;  
 high = a;  
 }  
 for (int i = 1; i < strlen(str) - 1; i++)

{  
 if (str[i - 1] < str[i] && str[i + 1] > str[i]) res++;  
 }  
 return res;  
}  
  
 int main()

{  
 char str1[] = { "nagkyuwezgbm" };  
 static double arr[] = { 9,10,5,2,4,3,0,7,1,8 };  
 int b = sizeof(arr) / sizeof(double);  
 int t = func(str1);  
 res = 0;  
 int c = func(arr, b);  
 res = 0;  
 printf("Number from array: %d\nNumber from line: %d", c,t);  
 \_getch();  
 return 0;  
}



**Висновок** - я ознайомився з принципами перевантаження функцій.