Міністерство освіти і науки України

Національний університет „Львівська політехніка”

Кафедра “Електронних обчислювальних машин”



**Звіт з лабораторної роботи №1**

на тему:

“Перевантаження Функції”

**Виконав:**

Ст. гр. КІ-15

Нижньов М.Ю.

**Перевірив:**

Викладач

Козак Н.Б.

Львів – 2020

**Мета роботи:** познайомитися із перевантаженням функцій.

**Теоретичні відомості:**

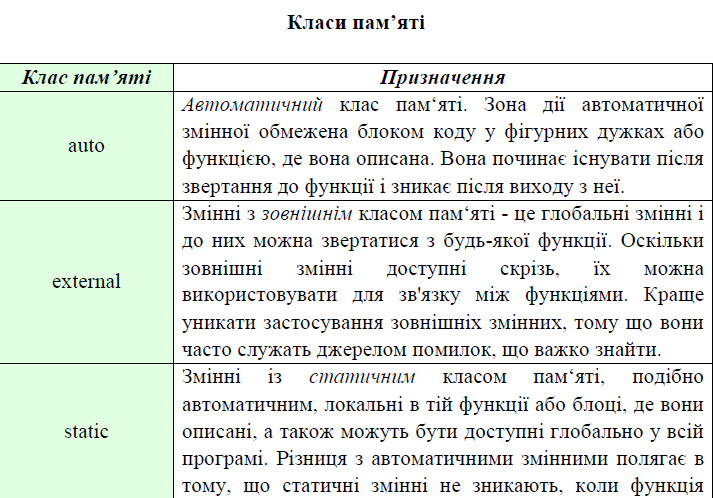
У мові С++, на відміну від мови С дозволяється визначати декілька функцій з одним і тим же іменем за умови, що дані функції мають різну сигнатуру (різні типи та кількість аргументів функції). При цьому тип значення що повертається при перевантаженні до уваги не приймається.

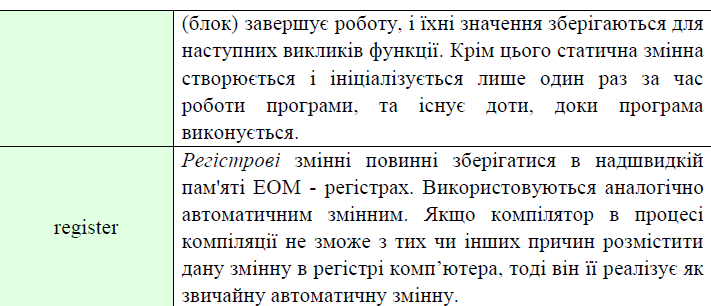
При виклику функції з ім‘ям func спершу компілятор намагатиметься знайти функцію, формальні аргументи якої відповідають фактичним аргументам без усяких перетворень типів або з використанням тільки неминучих перетворень - наприклад, імені масиву до покажчика або значення змінної до константи або навпаки.

Якщо відповідна функція не знайдена, то здійснюється пошук такої функції, щоб для відповідності формальних і фактичних аргументів досить було використати тільки такі стандартні перетворення, що не спричиняють перетворень цілих типів до типів з плаваючою крапкою і навпаки. При цьому підбирається функція, для якої число таких перетворень було б мінімальним.

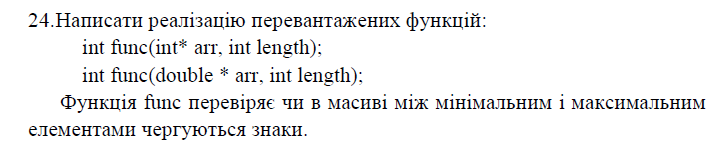
Третім етапом є пошук такої функції, для виклику якої досить здійснити будь-які стандартні перетворення аргументів (і знову так, щоб цих перетворень було якнайменше).

Далі здійснюється пошук функції, для якої аргументи можна одержати за допомогою всіх перетворень, розглянутих до цього, а також перетворень типів, визначених самим програмістом. Якщо й у цьому випадку єдина потрібна функція не знайдена, то на останньому етапі компілятор пробує знайти відповідність з урахуванням списку невизначених аргументів. Якщо компілятор не знайде жодної підходящої функції, або виклик функції не може бути однозначно співставлений з однією з оголошених функцій, то програма не скомпілюється і буде виведене повідомлення про помилку.





***Завдання:***



*Код програми:*

#include <stdio.h>

int const N=5;

int func(int\* arr, int len)

{

register int max\_i = 0, min\_i = 0, value=0, flag=1;

int max = arr[0], min = arr[1];

for (int i = 0; i < len; i++) //шукає макс. і мін. знач.

{

if (arr[i] > max)

{

max = arr[i];

max\_i = i;

}

if (arr[i] < min)

{

min = arr[i];

min\_i = i;

}

}

if (max\_i < min\_i)

{

min\_i = value;

min\_i = max\_i;

max\_i = value;

}

if (arr[min\_i] < 0)

{

flag = 0;

}

else flag = 1;

for (int i = (min\_i + 1); i < max\_i; i++)

{

if (flag==1 & arr[i] < 0)

{

flag = 0;

continue;

}

if (flag == 0 & arr[i] >= 0)

{

flag = 1;

continue;

}

if ((flag==1 & arr[i]>=0)||(flag==0 & arr[i]<0))

{

printf("Integer arr error!");

goto finish;

}

}

printf("Integer arr no error.");

finish:

return 0;

}

int func(double\* arr, int len)

{

register int max\_i = 0, min\_i = 0, value=0, flag = 1;

double max = arr[0], min = arr[1];

for (int i = 0; i < len; i++)

{

if (arr[i] > max)

{

max = arr[i];

max\_i = i;

}

if (arr[i] < min)

{

min = arr[i];

min\_i = i;

}

}

if (max\_i < min\_i)

{

min\_i = value;

min\_i = max\_i;

max\_i = value;

}

if (arr[min\_i] < 0)

{

flag = 0;

}

else flag = 1;

for (int i = min\_i + 1; i < max\_i; i++)

{

if (flag == 1 & arr[i] < 0)

{

flag = 0;

continue;

}

if (flag == 0 & arr[i] >= 0)

{

flag = 1;

continue;

}

if ((flag == 1 & arr[i] >= 0) || (flag == 0 & arr[i] < 0))

{

printf("Double arr error!");

goto finish;

}

}

printf("Double arr no error.");

finish:

return 0;

}

int main()

{

int a1[N];

double a2[N];

for (int i = 0; i < N; i++)

{

printf("Enter your integer num:");

scanf\_s("%d", &a1[i]);

}

printf("\n");

for (int i = 0; i < N; i++)

{

printf("Enter your double num:");

scanf\_s("%lf", &a2[i]);

}

func(a1, N);

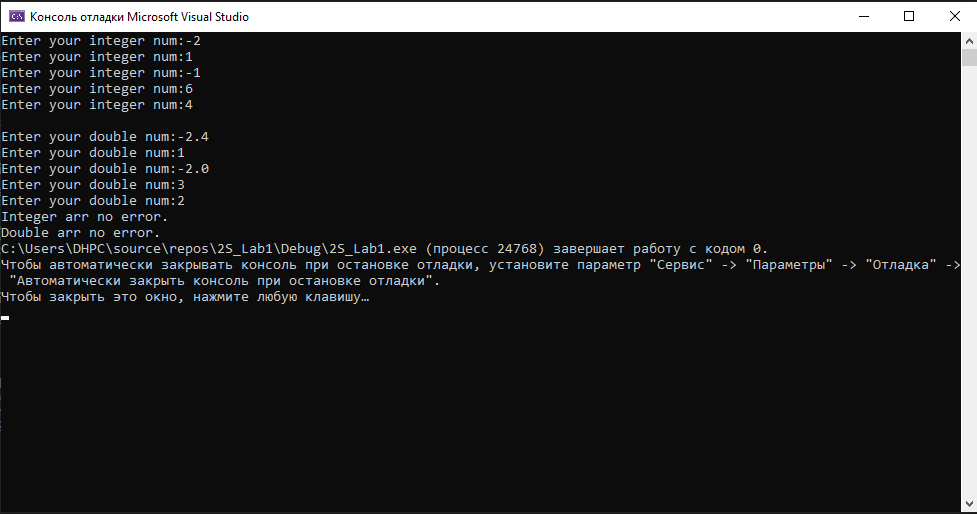
printf("\n");

func(a2, N);

return 0;

}

*Вікно результату:*

**

*Висновок:* я познайомився з перевантаженням функції та з класами памяті. Також навчився їх використовувати*.*