Міністерство освіти і науки України

Національний університет «Львівська політехніка»



**Лабораторна робота №2**

з дисципліни “Системне програмування”

на тему “Змішане програмування на мовах С та Асемблер”

Варіант 22

Виконав: ст. гр. КІ-38

Шашурин В. С.

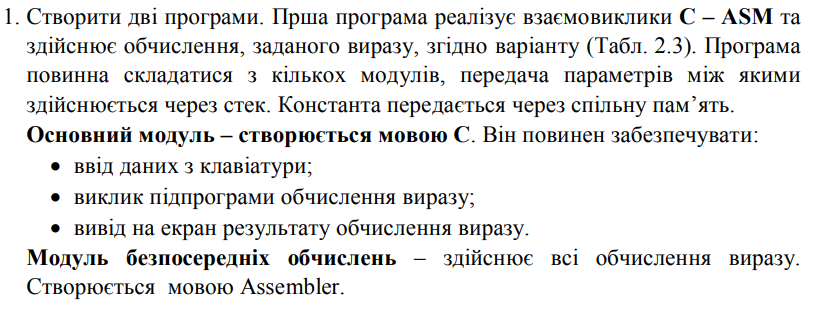
Прийняв:

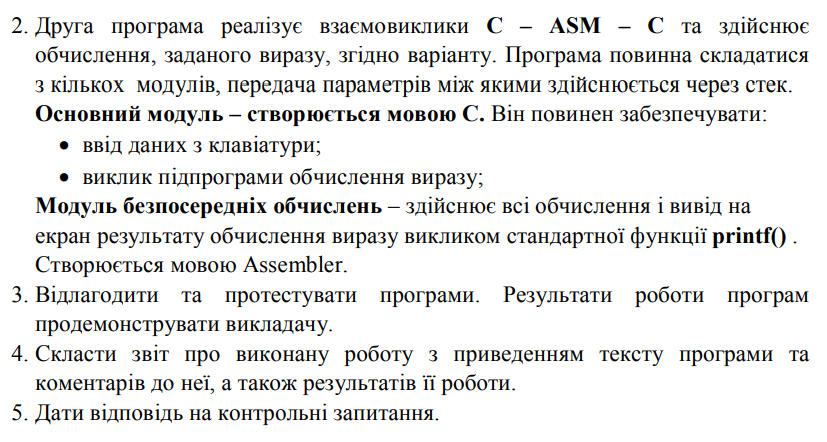
Асистент Козак Н. Б.

Львів – 2020

**Мета:** оволодіти навичками створення програм, частини яких написані різними мовами програмування. Засвоїти правила взаємодії між різними програмними модулями

**Завдання:**





**Варіант завдання:**



**Код програми до завдання 1:**

**main.cpp**

#include <stdio.h>

extern "C" int calc(signed char, signed char, short);

int main()

{

signed char b = 0;

signed char c = 0;

short d = 0;

int res = 0;

printf("Enter numbers:\n");

printf("B = ");

scanf\_s("%hhd", &b);

printf("C = ");

scanf\_s("%hhd", &c);

printf("D = ");

scanf\_s("%hd", &d);

res = calc(b, c, d);

printf("Result is: %d", res);

return 0;

}

**calc.asm**

.386

.model flat, c

.data

K equ 45781022h

div\_val db 0

mul\_val dw 0

.code

calc proc

push ebp

mov ebp, esp

xor eax, eax

mov ax, word ptr [ebp+16]

mov bl, 2

idiv bl

mov div\_val, al

xor eax, eax

mov al, byte ptr [ebp+8]

mov bl, 4

imul bl

mov mul\_val, ax

xor eax, eax

mov eax, K

sub eax, dword ptr [ebp+8]

sub eax, dword ptr [ebp+12]

sub al, div\_val

add ax, mul\_val

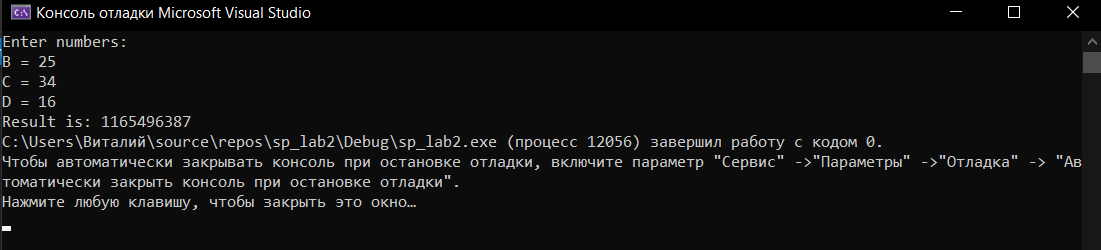
pop ebp

ret

calc endp

end

**Результат виконання:**



**Код програми до завдання 2:**

**main.cpp**

#include <stdio.h>

extern "C" int calc(signed char, signed char, short);

int main()

{

signed char b = 0;

signed char c = 0;

short d = 0;

int res = 0;

printf("Enter numbers:\n");

printf("B = ");

scanf\_s("%hhd", &b);

printf("C = ");

scanf\_s("%hhd", &c);

printf("D = ");

scanf\_s("%hd", &d);

res = calc(b, c, d);

return 0;

}

**calc.asm**

.386

.model flat, c

printf proto arg1:ptr byte, printlist: vararg

.data

K equ 45781022h

div\_val db 0

mul\_val dw 0

Msg db 'Result is: %d', 0

.code

calc proc

push ebp

mov ebp, esp

xor eax, eax

mov ax, word ptr [ebp+16]

mov bl, 2

idiv bl

mov div\_val, al

xor eax, eax

mov al, byte ptr [ebp+8]

mov bl, 4

imul bl

mov mul\_val, ax

xor eax, eax

mov eax, K

sub eax, dword ptr [ebp+8]

sub eax, dword ptr [ebp+12]

sub al, div\_val

add ax, mul\_val

pop ebp

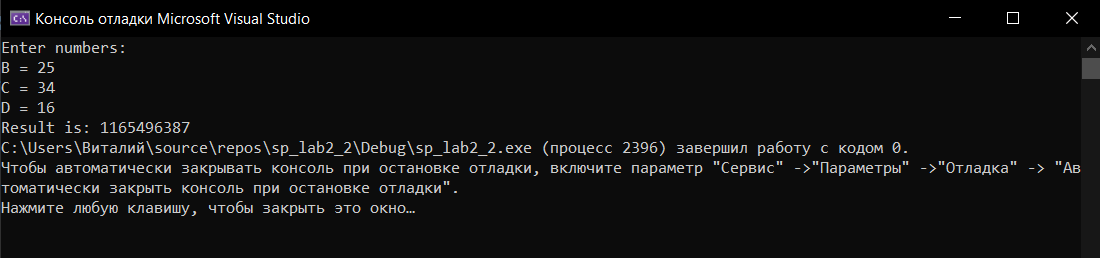
invoke printf, addr Msg, eax

ret

calc endp

end

**Результат виконання:**



**Висновок:** на даній лабораторній роботі я оволодів навичками створення програм, частини яких написані різними мовами програмування, та засвоїв принципи взаємодії їх модулів.