**Міністерство освіти і науки України**

**Національний університет «Львівська політехніка»**



Звіт

до лабораторної роботи № 2

з дисципліни: «Системне програмування»

«Змішане програмування на мовах С та Асемблер»

Виконав:

ст. гр. КІ-38

Романюк Т. М.

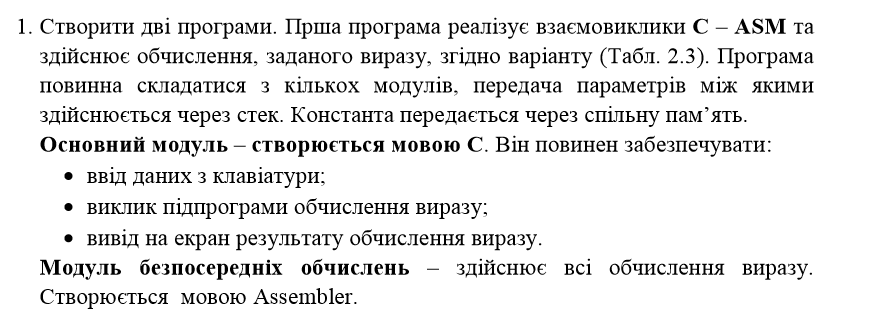
Прийняв:

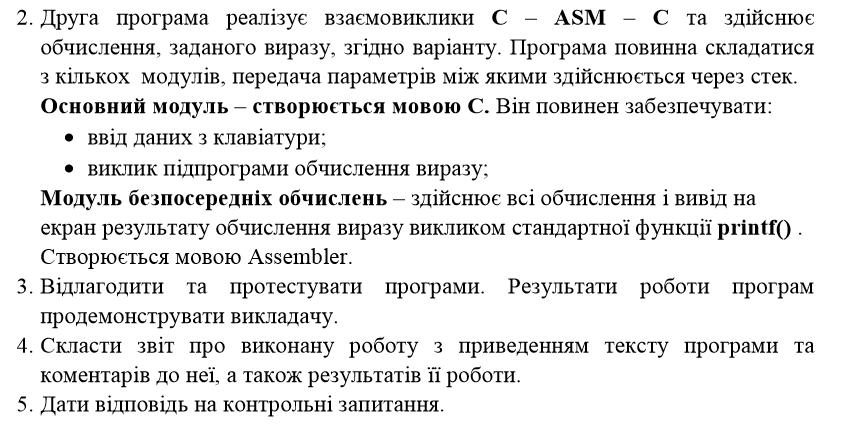
Козак Н. Б.

**Львів 2020**

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

**ЗАВДАННЯ:**





Варіант №14 

**Виконання:**

Код програми 1 (C - ASM)

***Lab2.cpp***

#include <stdio.h>

#include <windows.h>

#include <iostream>

extern "C" int calc(int j, char l, int h);

void main(void)

{

int a = 0, B; // 4

char b = 0; // 1

short int d = 0; // 2

int res = 0;

std::cout << "Enter A:" << std::endl;

std::cin >> a;

std::cout << "Enter B:" << std::endl;

std::cin >> B;

std::cout << "Enter D:" << std::endl;

std::cin >> d;

b = B;

res = calc(a, b, d);

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

}

***source.asm***

.586

.model flat

;EXTRN \_K:dword

PUBLIC \_calc

.data

K dd 614h

Two db 2

Four db 4

.code

; Процедура \_calc має 3 параметри і використовує узгодження мови С

\_calc PROC

; в стеку на даний момент є 32-ох розрядна ближня адреса пвернення з процедури

; збережемо в стек 32-ох розрядне значення ebp

push ebp

mov ebp, esp

; завантажити перший параметр який має відносно esp зміщення +8 байт

mov ebx, [ebp+8]; A

; завантажити другий параметр який має відносно esp зміщення +12 байт

mov al, [ebp+12]; B

cbw

cwd

sub ebx, eax

sub ebx, K

; завантажити третій параметр який має відносно esp зміщення +16 байт

mov eax, [ebp+16]; D

div Two

sub ebx, eax

; завантажити другий параметр який має відносно esp зміщення +12 байт

mov eax, [ebp+12]; B

cbw

cwd

mul Four

add ebx, eax

mov eax, ebx

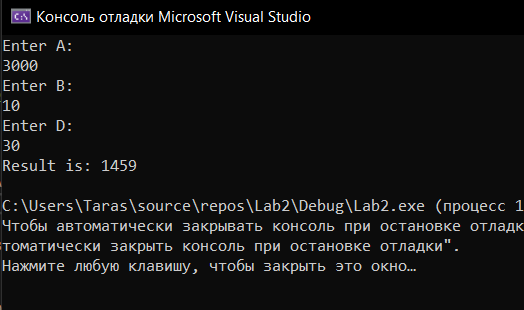
pop ebp

ret

\_calc endp

end

Результат виконання програми:



Код програми 2 (C – ASM - С)

***Lab2\_2.cpp***

#include <stdio.h>

#include <windows.h>

#include <iostream>

extern "C" int calc(int j, char l, int h);

extern "C" void print(int arg1)

{

printf("Result is: %d\n", arg1);

}

void main(void)

{

int a = 0, B; // 4 байти

char b = 0; // 1 байт

short int d = 0; // 2 байти

int res = 0;

std::cout << "Enter A:" << std::endl;

std::cin >> a;

std::cout << "Enter B:" << std::endl;

std::cin >> B;

std::cout << "Enter D:" << std::endl;

std::cin >> d;

b = B;

res = calc(a, b, d);

}

***source.asm***

.586

.model flat

;EXTRN \_K:dword

EXTRN \_print:PROC

PUBLIC \_calc

.data

K dd 614h

Two db 2

Four db 4

.code

; Процедура \_calc має 3 параметри і використовує узгодження мови С

\_calc PROC

; в стеку на даний момент є 32-ох розрядна ближня адреса пвернення з процедури

; збережемо в стек 32-ох розрядне значення ebp

push ebp

mov ebp, esp

; завантажити перший параметр який має відносно esp зміщення +8 байт

mov ebx, [ebp+8]; A

; завантажити другий параметр який має відносно esp зміщення +12 байт

mov al, [ebp+12]; B

cbw

cwd

sub ebx, eax

sub ebx, K

; завантажити третій параметр який має відносно esp зміщення +8 байт

mov eax, [ebp+16]; D

div Two

sub ebx, eax

; завантажити другий параметр який має відносно esp зміщення +8 байт

mov eax, [ebp+12]; B

cbw

cwd

mul Four

add ebx, eax

mov eax, ebx

; занести параметри функції в стек (arg1)

push eax

; викликати С функцію print

call \_print

; видалити параметри print зі стеку додаванням до esp числа 4

add esp,4

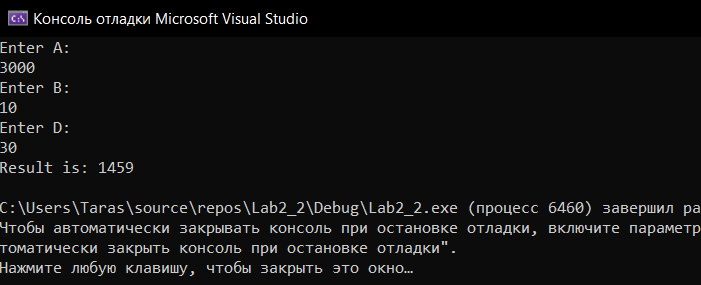
pop ebp

ret

\_calc endp

end

Результат виконання програми:



**Висновок:** на даній лабораторній роботі яоволодів навиками створення програм, частини яких написані різними мовами програмування. Також засвоїв правила взаємодії між різними програмними модулями.