Міністерство освіти і науки України

Національний університет “Львівська політехніка”



**Звіт**

З лабораторної роботи №2

З дисципліни: «Системне програмування»

На тему: «Змішане прошрамування на мовах С та Ассемблер»

Виконав: ст. гр. КІ-38

Черкасов В.С.

Прийняв: Козак Н.Б.

Львів 2020

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

**Завдання:** створити програму, яка реалізовує обчислення, заданого виразу, згідно варіанту. Програма повинна складатися з кількох модулів, передача параметрів між якими здійснюється через стек. Основний модуль – створюється мовою С. Він повинен забезпечувати:

ввід даних з клавіатури;

виклик підпрограми обчислення виразу;

використати взаємовиклики С – ASM – С.

Для виводу результатів обчислень забезпечити виклик стандартної функції printf() із модуля ASM. Модуль безпосередніх обчислень – здійснює всі обчислення виразу і створюється мовою Assembler; Відлагодити та протестувати програму. Результати роботи програми продемонструвати викладачу. Скласти звіт про виконану роботу з приведенням тексту програми та коментарів до неї, а також результатів її роботи.

А, В, С, D, E, F - знакові операнди, довжиною в байтах, згідно з індексу,

значення К подано у 16-му форматі.

Варіант завдання:

****

Завдання 1:

**Код програми:**

#include <stdio.h>

extern "C" int calc(short C, int A, long D);

void main(void)

{

int nRes, nNumber2, nNumber3, nNumber4;

printf("C = "); scanf\_s("%d", &nNumber2);

printf("D = "); scanf\_s("%d", &nNumber3);

printf("A = "); scanf\_s("%d", &nNumber4);

nRes = calc(nNumber2, nNumber3, nNumber4);

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

}

.386

.model flat, c

PUBLIC \_calc

.data

K EQU 4019h

Temp1 dd 00h

Temp2 dd 00h

.code

\_calc PROC

push ebp

mov ebp,esp

xor eax,eax

xor ebx,ebx

xor ecx,ecx

xor edx,edx

mov eax,[ebp+12]

mov bx, 2

idiv bx

mov Temp1,eax

mov eax,[ebp+16]

mov bx, 4

imul bx

sub eax, Temp1

add eax, [ebp+8]

add eax, K

pop ebp

ret

\_calc endp

end

Завдання 2:

#include <stdio.h>

extern "C" int calc(short C, int A, long D);

int main()

{

int nRes, nNumber2, nNumber3, nNumber4;

printf("C = "); scanf\_s("%d", &nNumber2);

printf("D = "); scanf\_s("%d", &nNumber3);

printf("A = "); scanf\_s("%d", &nNumber4);

nRes = calc(nNumber2, nNumber3, nNumber4);

return 0;

}

.386

.model flat, c

printf proto arg1:ptr byte, printlist: vararg

.data

K equ 4019h

Temp1 dd 0

Temp2 dd 0

Msg db 'Result is: %d', 0

.code

calc proc

push ebp

mov ebp,esp

xor eax,eax

xor ebx,ebx

xor ecx,ecx

xor edx,edx

mov eax,[ebp+12]

mov bx, 2

idiv bx

mov Temp1,eax

mov eax,[ebp+16]

mov bx, 4

imul bx

sub eax, Temp1

add eax, [ebp+8]

add eax, K

pop ebp

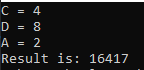
invoke printf, addr Msg, eax

ret

calc endp

end

Результати виконання програми:

****

2\*4+4- 8/2 + 4019 = 8 + 4019 = 16417

**Висновок:** на даній лабораторній роботі я опанував роботу з програмною моделлю 32 розрядних процесорів Intel та оволодів навиками створення програм, використовуючи 32 розрядний асемблер.