Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки

**ЛАБОРАТОРНА РОБОТА №5**

з дисципліни «Системне програмування» на тему

«Арифметичні і логічні операції з цілими числами. Масиви»

Виконав: Перевірив:

студент II курсу ФІОТ доц. Павлов В. Г. групи ІМ-22 Басараб Станіслав Анатолійович  
номер у списку групи: 2

Київ 2024

**Мета роботи**: Вивчення арифметичних і логічних команд Асемблера і здобуття навиків виконання розрахунків з елементами масивів.

**Порядок виконання роботи**

1. Вивчити арифметичні і логічні команди цілочисельної арифметики.
2. Розробити програму на мові Асемблер, в якій згідно з індивідуальним варіантом завдання виконуються обчислення значення арифметичного виразу з подальшим виводом результату у віконному інтерфейсі.
3. Для всіх варіантів: в разі парного результату він перед виводом додатково ділиться на 2, а в разі непарного – результат додатково умножається на 5.
4. Розрахунки (п. 2, 3) повторити в програмі для 5 значень змінних, причому всі вихідні значення задати цілими числами у вигляді одновимірних масивів.
5. Для перевірки правильності виконання розрахунків і результатів, що виводяться, заздалегідь виконати контрольні розрахунки. Проміжні і остаточні результати контрольних розрахунків привести в звіті по лабораторній роботі.
6. Виконати відладку програми шляхом порівняння розрахованих програмою результатів з контрольними прикладами. Лістинг розробленої програми і скріншоти розрахунків по всіх контрольних прикладах привести в звіті по лабораторній роботі.
7. Зробити висновки по лабораторній роботі.

**Виконання роботи**

Формула для розрахунку за 2 варіантом: (-25 / a + c – b \* a) / (l + c \* b/ 2).

Контрольні розрахунки

1. Знаменник і чисельник > 0

a = -25  
b = 2  
c = 4

(-25 / (-25) + 4 – 2 \* (-25)) / (1 + 4 \* 2 / 2) = (1 + 4 + 50) / (1 + 4) =   
55 / 5 = 11

Результат – непарний, множимо на 5

11 \* 5 = 55

Проміжний результат = 11

Остаточний результат = 55

1. Чисельник < 0, а знаменник > 0

a = 5

b = 2

c = 3

(-25 / 5 + 3 – 2 \* 5) / (1 + 3 \* 2 / 2) = (-5 + 3 - 10) / (1 + 3) = -12 / 4 = -3

Результат – непарний, множимо на 5

-3 \* 5 = -15

Проміжний результат = -3

Остаточний результат = -15

1. Чисельник > 0, а знаменник < 0

a = -5

b = 6

c = -9

(-25 / (-5) + (-9) – 6 \* (-5)) / (1 + (-9) \* 6 / 2) = (5 – 9 + 30) / (1 - 27) =  
26 / (-26) = -1

Результат – непарний, множимо на 5

-1 \* 5 = -5

Проміжний результат = -1

Остаточний результат = -5

1. Знаменник і чисельник < 0

a = 25

b = 8

c = -2

(-25 / 25 + (-2) – 8 \* 25) / (1 + (-2) \* 8 / 2) = (-1 - 2 - 200) / (1 - 8) =  
-203 / (-7) = 29

Результат – непарний, множимо на 5

29 \* 5 = 145

Проміжний результат = 29

Остаточний результат = 145

1. Знаменник = 0

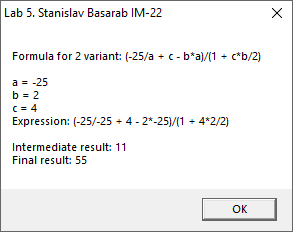
a = 5

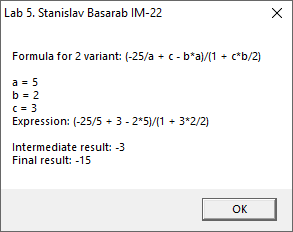
b = 2

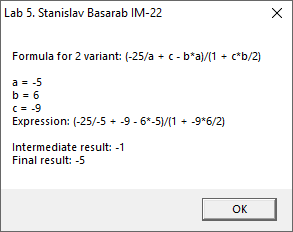
c = -1

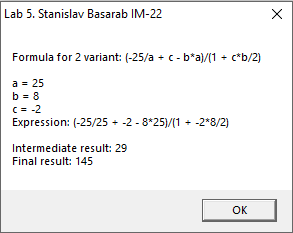
(-25 / 5 + (-1) – 2 \* 5) / (1 + (-1) \* 2 / 2) = (-5 - 1 - 10) / (1 - 1) = -16 / 0

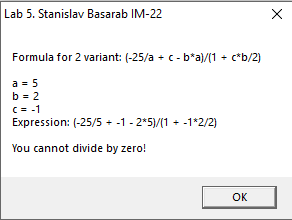
Робота програми











Тож, порівнюючи контрольні розрахунки з результатами виконання програми, можна стверджувати, що програма працює абсолютно правильно в усіх випадках:

* Знаменник і чисельник > 0
* Чисельник < 0, а знаменник > 0
* Чисельник > 0, а знаменник < 0
* Знаменник і чисельник < 0
* Знаменник = 0

Лістинг програми

.386

.model flat, stdcall

option casemap:none

include \masm32\include\masm32rt.inc

.data?

SBasarabWindowContentB db 256 dup (?)

SBasarabResultContentB db 256 dup (?)

SBasarabDown dd ?

SBasarabResult dd ?

.data

SBasarabArrayForA dd -25, 5, -5, 25, 5

SBasarabArrayForB dd 2, 2, 6, 8, 2

SBasarabArrayForC dd 4, 3, -9, -2, -1

SBasarabWindowCaption db "Lab 5. Stanislav Basarab IM-22", 0

SBasarabMainWindowContent db "Formula for 2 variant: (-25/a + c - b\*a)/(1 + c\*b/2)", 10, 10,

"a = %d", 10,

"b = %d", 10,

"c = %d", 10,

"Expression: (-25/%d + %d - %d\*%d)/(1 + %d\*%d/2)", 10, 10,

"%s", 0

SBasarabResultWindowContent db "Intermediate result: %d", 10,

"Final result: %d", 0

SBasarabErrorWindowContent db "You cannot divide by zero!"

SBasarabCounter dd 0

.code

start:

mov esi, SBasarabCounter

start\_loop\_label:

mov eax, SBasarabArrayForB[4 \* esi]

mov ebx, 2

cdq

idiv ebx

mov ecx, SBasarabArrayForC[4 \* esi]

imul eax, ecx

add eax, 1

.if eax == 0

invoke wsprintf, offset SBasarabWindowContentB, offset SBasarabMainWindowContent,

SBasarabArrayForA[4 \* esi], SBasarabArrayForB[4 \* esi], SBasarabArrayForC[4 \* esi],

SBasarabArrayForA[4 \* esi], SBasarabArrayForC[4 \* esi], SBasarabArrayForB[4 \* esi], SBasarabArrayForA[4 \* esi],

SBasarabArrayForC[4 \* esi], SBasarabArrayForB[4 \* esi], offset SBasarabErrorWindowContent

.else

mov SBasarabDown, eax

mov eax, -25

mov ebx, SBasarabArrayForA[4 \* esi]

cdq

idiv ebx

mov edx, SBasarabArrayForB[4 \* esi]

imul edx, ebx

add eax, ecx

sub eax, edx

mov ebx, SBasarabDown

cdq

idiv ebx

mov SBasarabResult, eax

test eax, 1

jz processEven

jnz processOdd

display:

invoke wsprintf, offset SBasarabResultContentB, offset SBasarabResultWindowContent,

SBasarabResult, eax

invoke wsprintf, offset SBasarabWindowContentB, offset SBasarabMainWindowContent,

SBasarabArrayForA[4 \* esi], SBasarabArrayForB[4 \* esi], SBasarabArrayForC[4 \* esi],

SBasarabArrayForA[4 \* esi], SBasarabArrayForC[4 \* esi], SBasarabArrayForB[4 \* esi], SBasarabArrayForA[4 \* esi],

SBasarabArrayForC[4 \* esi], SBasarabArrayForB[4 \* esi], offset SBasarabResultContentB

.endif

invoke MessageBox, 0, offset SBasarabWindowContentB, offset SBasarabWindowCaption, 0

inc esi

.if esi == 5

invoke ExitProcess, 0

.endif

jmp start\_loop\_label

processEven:

mov ebx, 2

cdq

idiv ebx

jmp display

processOdd:

mov ebx, 5

imul eax, ebx

jmp display

end start

Висновок

Отже, під час виконання лабораторної роботи, я вивчив арифметичні і логічні команди Асемблера і здобув навики виконання розрахунків з елементами масивів. Були проведені контрольні розрахунки для 5 різних випадків, а також написана програма, яка у віконному режимі демонструє ці розрахунки. Було порівняно контрольні розрахунки з результатами виконання програми:

* Знаменник і чисельник > 0
* Чисельник < 0, а знаменник > 0
* Чисельник > 0, а знаменник < 0
* Знаменник і чисельник < 0
* Знаменник = 0

Усі отримані значення з програми абсолютно співпадають з розрахунками. Це говорить про правильність роботи програми.