**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 групи ІП-05 доцент Павлов В.Г.

Кононенко Вадим Олександрович

номер залікової книжки: 0512

Варіант 13

Київ-2022

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

**Рівнянння:** ((c/4)+(28\*d)) / ((a/d)-c-1)

**Контрольні розрахунки:**

1. a: 81 c: 8 d: 9  
   (8/4 + 28\*9) / (81/9 – 8 - 1) = (2 + 252)/(9 – 8 - 1) – помилка ділення на 0
2. a: 55 c: 3 d: 4  
   (3/4 + 28\*4) / (55/4 – 3 - 1)= (-140 + 164)/(25-1) – помилка не ціле число
3. a: 76 c: 8 d: 2  
   (8/4 + 28\*2) / (76/2 – 8 - 1) = (2 + 56) / (38 – 8 - 1) = 58 / 29 = 2

Оскільки парне, то фінальна відповідь: **1**

1. a: -32 c: 4 d: -1  
   (4/4 + 28\*(-1)) / ((-32)/(-1) – 4 - 1) = (1 - 28) / (32 – 4 - 1) = -27 / 27 = -1

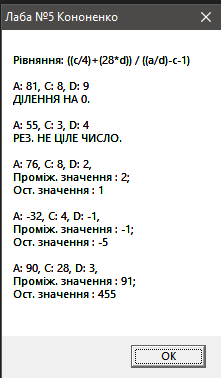
Оскільки непарне, то фінальна відповідь: **-5**

1. a: 90 c: 28 d: 3

(28/4 + 28\*3) / (90/3 – 28 - 1) = (7 + 84) / (30 – 28 - 1) = 91 / 1 = 91

Оскільки непарне, то фінальна відповідь: 455

Скріншот виконання програми



**Код програми:**

.686

.model flat, stdcall

option casemap:none

include <\masm32\include\windows.inc>

include <\masm32\include\dialogs.inc>

include <\masm32\include\user32.inc>

includelib <\masm32\lib\user32.lib>

includelib <\masm32\lib\masm32.lib>

include <\masm32\include\masm32.inc>

.data

header db 'Лаба №5 Кононенко', 0

funcTask db 'Рівняння: ((c/4)+(28\*d)) / ((a/d)-c-1)',13,13,0

outputRes db "А: %d, С: %d, D: %d,",13,"Проміж. значення : %d;",13,"Ост. значення : %d",13,13,0

outputErrZero db "А: %d, С: %d, D: %d",13,"ДІЛЕННЯ НА 0.",13,13,0

outputErrRem db "А: %d, С: %d, D: %d",13,"РЕЗ. НЕ ЦІЛЕ ЧИСЛО.",13,13,0

divVal dd 4

inAValue dd 81, 55, 76, -32, 90

inCValue dd 8, 3, 8, 4, 28

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

buffer db 86 dup(?)

outputVal db 512 dup(?)

OResVal dd 5 dup(?)

FResVal dd 5 dup(?)

.code

start:

invoke wsprintf, addr outputVal, addr funcTask

mov edi, 0

.WHILE edi <5

mov eax, inAValue[4\*edi]

mov ebx, inDValue[4\*edi]

mov ecx, inCValue[4\*edi]

cmp ebx, 0

je errorZero

cdq

idiv ebx

mov edx,eax

imul edx, ebx

cmp edx, inAValue[4\*edi]

jne errRem

sub eax, ecx

dec eax

cmp eax, 0

je errorZero

mov esi, eax

mov eax, ecx

cdq

idiv divVal

mov edx, eax

imul edx, divVal

cmp edx, ecx

jne errRem

imul ebx, 28

add eax, ebx

mov ecx, eax

cdq

idiv esi

mov edx, eax

imul edx, esi

cmp edx, ecx

jne errRem

mov FResVal[4\*edi], eax

test eax, 1

jnz odd

mov esi, 2

cdq

idiv esi

jmp outif

odd:

imul eax, 5

outif:

mov OResVal[4\*edi], eax

invoke wsprintf, addr buffer, addr outputRes, inAValue[4\*edi], inCValue[4\*edi], inDValue[4\*edi], FResVal[4\*edi], OResVal[4\*edi]

jmp cont

errorZero:

invoke wsprintf, addr buffer, addr outputErrZero, inAValue[4\*edi], inCValue[4\*edi], inDValue[4\*edi]

jmp cont

errRem:

invoke wsprintf, addr buffer, addr outputErrRem, inAValue[4\*edi], inCValue[4\*edi], inDValue[4\*edi]

jmp cont

cont:

invoke szCatStr, addr outputVal, addr buffer

inc edi

.ENDW

invoke MessageBox, 0, addr outputVal, addr header, 0

end start

**Висновки**

Під час виконання цієї лабораторної роботи я вдосконалив свої знання в арифметичних та логічних командах masm32, реалізував програму яка обчислює певний вираз та виводить його на екран. Проаналізував дані, порівняв вивід та контрольні розрахунки і впевнився, що програма робить коректно.