**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 №7

з дисципліни

«Системне програмування»

на тему

«Модульне програмування. Використання процедур. »

Виконала: Перевірив:

Студентка групи ІМ-21 доцент Павлов В. Г.

Дворецька Анастасія Вадимівна

Номер у списку групи: 6

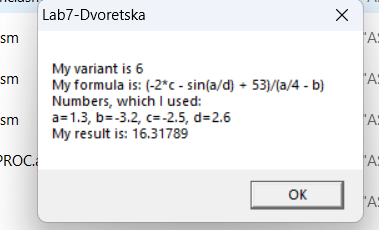
**Київ 2024**

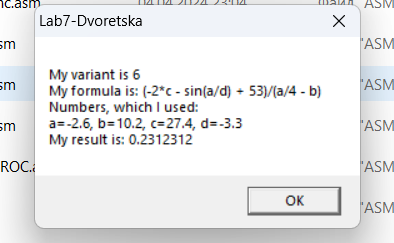
Варіант 6

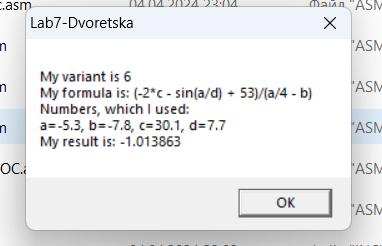
**Мета:**

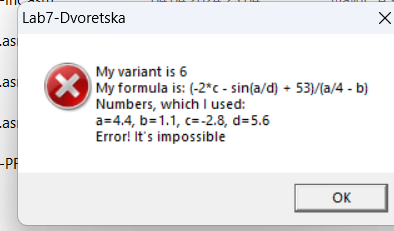
Вивчення прийомів модульного програмування, методів звернення до процедур і передачі в них параметрів.

**1.Скріншоти виконання програми:**

****

****

****

****

**2.Конрольні розрахунки:**

**Формула:**

1. У знаменнику й чисельнику додатні числа:

a = 1.3, b = -3.2, c = -2.5, d = 2.6

(-2 \* (-2.5) - sin(1.3/2.6) + 53)/(1.3 / 4 + 3.2) = (5 - sin(0.5) + 53)/(0.325 + 3.2) = (58 - 0.479426)/(3.525) = 57.520574/3.525 = **16.317893**

1. У знаменнику й чисельнику від’ємні числа:

a = -2.6, b = 10.2, c = 27.4, d = -3.3

(-2 \* 27.4 - sin(-2.6/-3.3) + 53)/(-2.6 / 4 - 10.2) = (-54.8 - sin(0.787879) + 53)/(-0.65 - 10.2) = (-1.8 - 0.708859)/(-10.85) = -2.508859/-10.85 = **0.23123124**

1. У знаменнику-додатнє,у чисельнику-від’ємне:

a = -5.3, b = -7.8, c = 30.1, d = 7.7

(-2 \* 30.1 - sin(-5.3/7.7) + 53)/(-5.3 / 4 + 7.7) = (-60.2 + sin(0.688312) + 53)/(-1.325 + 7.8) = (-7.2 + 0.688312)/(6.475) = -6.564766/6.475 = **-1.013863**

1. У знаменнику-від’ємне,у чисельнику-додатнє:

a = 4.1, b = 4.3, c = -5.9, d = 7.3

(-2 \* (-5.9) - sin(4.1/7.3) + 53)/(4.1 / 4 - 4.3) = (11.8 - sin(0.561644) + 53)/(1.025 - 4.3) = (64.8 - 0.532578)/(-3.275) = 64.267422/-3.275 = **-19.623640**

1. Знаменник дорівнює 0:

a = 4.4, b = 1.1, c = -2.8, d = 5.6

(-2 \* (-2.8) - sin(4.4 / 5.6) + 53)/(4.4 / 4 - 1.1) = (5.6 - sin(0.785712) + 53)/(1.1 - 1.1) = (68.6 - sin(0.785712))/0 => **Повідомлення про помилку**

**Інших випадків немає,адже область визначення функції синус не має обмежень**

**3.Лістинг програми:**

.386

.model flat, stdcall

option casemap :none

include \masm32\include\masm32rt.inc

public Array\_Of\_My\_A, Array\_Of\_My\_B

extern my\_div2\_result:PROTO, my\_dec2\_result:PROTO

.data

buff db 512 dup(?)

buff\_of\_null db 512 dup(?)

buff\_of\_error db 512 dup(?)

buff\_of\_a db 256 dup(?)

buff\_of\_b db 256 dup(?)

buff\_of\_c db 256 dup(?)

buff\_of\_d db 256 dup(?)

buff\_of\_res db 512 dup(?)

Array\_Of\_My\_A dq 1.3, -2.6, 4.1, -5.3, 4.4

Array\_Of\_My\_B dq -3.2, 10.2, 4.3, -7.8, 1.1

Array\_Of\_My\_C dq -2.5, 27.4, -5.9, 30.1, -2.8

Array\_Of\_My\_D dq 2.6, -3.3, 7.3, 7.7, 5.6

title\_of\_my\_messagebox db "Lab7-Dvoretska",0

data\_of\_my\_messagebox db "My variant is 6",10

db "My formula is: (-2\*с - sin(a/d) + 53)/(a/4 - b)",10

db "Numbers, which I used:",10

db "a=%s, b=%s, c=%s, d=%s",10

db "My result is: %s",0

data\_of\_my\_error\_messagebox db "My variant is 6",10

db "My formula is: (-2\*с - sin(a/d) + 53)/(a/4 - b)",10

db "Numbers, which I used:",10

db "a=%s, b=%s, c=%s, d=%s",10

db "Error! It's impossible",0

;my\_current\_a db 256 dup(?)

;my\_current\_b db 256 dup(?)

;my\_current\_c db 256 dup(?)

;my\_current\_d db 256 dup(?)

my\_current\_a dq ?

my\_current\_b dq ?

my\_current\_c dq ?

my\_current\_d dq ?

my\_mul\_result dt ?

my\_div1\_result dt ?

my\_dec1\_result dt ?

my\_sin\_result dt ?

my\_add\_result dt ?

;my\_div2\_result dt ?

;my\_dec2\_result dt ?

my\_end\_result dq ?

my\_help dd ?

my\_test dq ?

.code

mainLab7:

xor esi,esi

xor edi,edi

xor ebp,ebp

xor ecx,ecx

xor eax,eax

finit

ffree st(0)

mov esi, OFFSET Array\_Of\_My\_A

mov eax, OFFSET Array\_Of\_My\_B

mov edi, OFFSET Array\_Of\_My\_C

mov ebp, OFFSET Array\_Of\_My\_D

mov ecx, 5

loop\_for\_my\_lab:

ffree st(0)

push ecx

push esi

push eax

push edi

push ebp

fld qword ptr[ebp]

fstp qword ptr [my\_current\_d]

fld qword ptr [edi]

fstp qword ptr [my\_current\_c]

fld qword ptr [eax]

fstp qword ptr [my\_current\_b]

fld qword ptr [esi]

fstp qword ptr [my\_current\_a]

mov ebx, -2

call my\_first\_part

fld tbyte ptr[my\_mul\_result]

fld tbyte ptr[my\_sin\_result]

call my\_second\_part

ffree st(0)

fld tbyte ptr[my\_dec2\_result]

fldz

fcomp st(1)

fstsw ax

sahf

jz is\_null

jmp end\_check

is\_null:

invoke FloatToStr, my\_current\_a, addr buff\_of\_a

invoke FloatToStr2, my\_current\_b, addr buff\_of\_b

invoke FloatToStr2, my\_current\_c, addr buff\_of\_c

invoke FloatToStr2, my\_current\_d, addr buff\_of\_d

invoke wsprintf, addr buff\_of\_null, addr data\_of\_my\_error\_messagebox, addr buff\_of\_a, addr buff\_of\_b, addr buff\_of\_c, addr buff\_of\_d

invoke MessageBox, 0, addr buff\_of\_null, addr title\_of\_my\_messagebox, MB\_ICONERROR

jz end\_of

end\_check:

ffree st(0)

fld tbyte ptr[my\_add\_result]

fld tbyte ptr[my\_dec2\_result]

fdiv

fstp qword ptr[my\_end\_result]

invoke FloatToStr, [my\_current\_a], addr buff\_of\_a

invoke FloatToStr2, [my\_current\_b], addr buff\_of\_b

invoke FloatToStr2, [my\_current\_c], addr buff\_of\_c

invoke FloatToStr, [my\_end\_result], addr buff\_of\_res

invoke FloatToStr2, [my\_current\_d], addr buff\_of\_d

invoke wsprintf, addr buff, addr data\_of\_my\_messagebox, addr buff\_of\_a, addr buff\_of\_b, addr buff\_of\_c, addr buff\_of\_d, addr buff\_of\_res

invoke MessageBox, 0, addr buff, addr title\_of\_my\_messagebox, 0

end\_of:

ffree st(0)

pop ebp

pop edi

pop eax

pop esi

add esi, 8

add eax, 8

add edi, 8

add ebp, 8

pop ecx

dec ecx

jnz loop\_for\_my\_lab

invoke ExitProcess, 0

my\_first\_part PROC

mov my\_help, ebx

fild dword ptr[my\_help]

fld qword ptr [my\_current\_c]

fmul st, st(1)

fstp tbyte ptr[my\_mul\_result]

fld qword ptr [my\_current\_a]

fld qword ptr [my\_current\_d]

fdiv

fstp tbyte ptr[my\_div1\_result]

fld tbyte ptr [my\_div1\_result]

fsin

fstp tbyte ptr[my\_sin\_result]

ret

my\_first\_part ENDP

my\_second\_part PROC

fsub

fstp tbyte ptr[my\_dec1\_result]

ffree st(0)

mov ebx, 53

mov my\_help, ebx

fld tbyte ptr[my\_dec1\_result]

fild dword ptr[my\_help]

fadd

fstp tbyte ptr[my\_add\_result]

ret

my\_second\_part ENDP

end mainLab7

**4.Лістинг з процедурою**

.386

.model flat, stdcall

option casemap :none

public my\_div2\_result, my\_dec2\_result

extern Array\_Of\_My\_A:qword, Array\_Of\_My\_B:qword

.code

my\_third\_part PROC

mov ebx, 4

mov my\_help, ebx

fld qword ptr [my\_current\_a]

fild dword ptr[my\_help]

fdiv

fstp tbyte ptr[my\_div2\_result]

fld tbyte ptr[my\_div2\_result]

fld qword ptr [my\_current\_b]

fsub

fstp tbyte ptr[my\_dec2\_result]

ret

my\_third\_part ENDP

end

**5.Висновок**

Протягом виконання лабораторної роботи,я більше дізналась про використання процедур у масм32,розробила три процедури із передаванням параметрів через регістри,стек і з використанням public тa extern.