**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 № 6**

з дисципліни

«Системне програмування»

на тему “Обчислення арифметичних виразів і трансцендентних функцій.

Використання команд співпроцесора іх87”

Виконав: Перевірив:

студент групи ІП-05 доцент Павлов В.Г.

Кононенко Вадим Олександрович

номер залікової книжки: 0512

Варіант 13

Київ-2022

**Мета роботи** : Вивчення команд Асемблера для арифметики з плаваючою комою і здобуття навичок виконання розрахунків з елементами масивів.

**Код програми**

include \masm32\include\masm32rt.inc

include \masm32\include\Fpu.inc

includelib \masm32\lib\Fpu.lib

main\_solver macro index

fld array\_a[8\*index]

fimul number2

fsub array\_c[8\*index]

fld array\_b[8\*index]

fidiv number3

fadd

fld array\_b[8\*index]

fld array\_d[8\*index]

fidiv number2

fsubp st(1), st

fld1

fpatan

fdivp st(1), st

fst solved

invoke crt\_sprintf, addr res\_solved\_values[100\*index], addr solved\_value\_template, solved

invoke crt\_sprintf, addr res\_given\_values[200\*index],

addr given\_values\_template,

array\_a[8\*index],

array\_b[8\*index],

array\_c[8\*index],

array\_d[8\*index]

endm

.data?

hin HINSTANCE ?

CommandLine LPSTR ?

solved dq ?

res\_given\_values db 1200 dup (?)

res\_solved\_values db 1200 dup (?)

.data

ClassName db "simple", 0

header db "Expression solver by Vadim Kononenko IP-05 Laba 6", 0

given\_values\_template db "A: ( %f )", 10,

" B: ( %f )", 10,

" C: ( %f )", 10,

" D: ( %f )", 0

solved\_value\_template db "(2a-c + b/3) / (arctg(b - d/2)) = ( %f )", 0

expression db "Expression: (2a-c + b/3) / (arctg(b - d/2))", 0

number2 dd 2

number3 dd 3

array\_a dq 25.12, 3.213, 796.5342, 40056321.4562, -0.0000023112

array\_b dq 15.43, 0.2381, 6.26, 4.87556, 4.4353111

array\_c dq 2.2, -184.0012, -7.42, -5421.62, -0.32411

array\_d dq 6.4, 18.84, 0.51, 1.03331, 0.45211

WinMain proto :dword,:dword,:dword,:dword

.code

start:

invoke GetModuleHandle, NULL

mov hin,eax

invoke GetCommandLine

mov CommandLine,eax

invoke WinMain, hin,NULL,CommandLine, SW\_SHOWDEFAULT

invoke ExitProcess, eax

WinMain proc hInst:HINSTANCE, hPrevInst:HINSTANCE, CmdLine:LPSTR, CmdShow:dword

local wcx:WNDCLASSEX

local msg:MSG

local hwnd:HWND

mov wcx.cbSize,SIZEOF WNDCLASSEX

mov wcx.style, CS\_HREDRAW or CS\_VREDRAW

mov wcx.lpfnWndProc, OFFSET WndProc

mov wcx.cbClsExtra,NULL

mov wcx.cbWndExtra,NULL

push hin

pop wcx.hInstance

mov wcx.hbrBackground,COLOR\_WINDOW+1

mov wcx.lpszMenuName,NULL

mov wcx.lpszClassName,OFFSET ClassName

invoke LoadIcon,NULL,IDI\_APPLICATION

mov wcx.hIcon,eax

mov wcx.hIconSm,eax

invoke LoadCursor,NULL,IDC\_ARROW

mov wcx.hCursor,eax

invoke RegisterClassEx, addr wcx

invoke CreateWindowEx, NULL,addr ClassName, addr header, WS\_OVERLAPPEDWINDOW, CW\_USEDEFAULT, CW\_USEDEFAULT,

600,

400,

NULL,

NULL,

hInst,

NULL

mov hwnd,eax

invoke ShowWindow, hwnd,CmdShow

invoke UpdateWindow, hwnd

.WHILE TRUE

invoke GetMessage, addr msg,NULL,0,0

.BREAK .IF (!eax)

invoke TranslateMessage, addr msg

invoke DispatchMessage, addr msg

.ENDW

mov eax,msg.wParam

ret

WinMain endp

WndProc proc hWnd:HWND, uMsg:UINT, wParam:WPARAM, lParam:LPARAM

local hdc:HDC

local ps:PAINTSTRUCT

.if uMsg==WM\_CREATE

counter = 5;

while counter;

counter = counter - 1

main\_solver counter

endm

.elseif uMsg==WM\_DESTROY

invoke PostQuitMessage,NULL

.elseif uMsg==WM\_PAINT

invoke BeginPaint, hWnd, addr ps

mov hdc, eax

invoke TextOut, hdc, 150, 20, addr expression, 43

invoke TextOut, hdc, 20, 60, addr res\_given\_values, 200

invoke TextOut, hdc, 20, 80, addr res\_solved\_values, 60

invoke TextOut, hdc, 20, 120, addr res\_given\_values[200], 200

invoke TextOut, hdc, 20, 140, addr res\_solved\_values[100], 60

invoke TextOut, hdc, 20, 180, addr res\_given\_values[400], 200

invoke TextOut, hdc, 20, 200, addr res\_solved\_values[200], 60

invoke TextOut, hdc, 20, 240, addr res\_given\_values[600], 200

invoke TextOut, hdc, 20, 260, addr res\_solved\_values[300], 60

invoke TextOut, hdc, 20, 300, addr res\_given\_values[800], 200

invoke TextOut, hdc, 20, 320, addr res\_solved\_values[400], 60

invoke EndPaint, hWnd, addr ps

.else

invoke DefWindowProc,hWnd,uMsg,wParam,lParam

ret

.endif

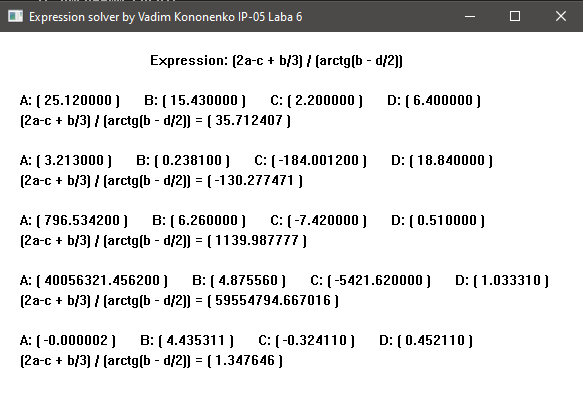
xor eax,eax

ret

WndProc endp

end start

**Результат роботи:**



**Контрольні розрахунки**

**First.**

**A:** (25,12) **B:** (15,43) **C:** (2,2) **D:** (6,4)

(2\*25,12 – 2,2 + 15,43/3) / arctg(15,43 – 6,4/2) = (50,24 – 2,2 + 1543/100\*1/3) / arctg(15,43 – 3,2) = (1201/25 + 1543/300) \* 1/arctg(12,23) = (3191/60) \* 1/arctg(12,23) = **35,71241**

**Second.**

**A:** (3,213) **B:** (0,2381) **C:** (-184,001200) **D:** (18,84)

(2\*3,213 + 184,0012 + 0,2381/3) / arctg(0,2381-18,84/2) = (190,4272 + 2381/30000) \* 1/arctg(0,2381 – 9,42) = (5715197/30000) \* (1 / -arctg(91819/10000)) = **-130,27747**

**Third.**

**A:** (796,5342) **B:** (6,26) **C:** (-7,42) **D:** (0,51)

(2\*796,5342 + 7,42 + 6,26/3) / arctg(6,26 – 0,51/2) = (1600,4884 + 313/150) \* 1/arctg(6,26 – 0,255) = (12019313/7500) \* 1 / arctg(1201/200) = **1139,98778**

**Fourth.**

**A:** (40056321,4562) **B:** (4,875560) **C:** (-5421,62) **D:** (1,033310)

(2\*40056321,4562 + 5421,62 + 4,87556/3) / arctg(4,87556 – 1,03331/2) = (200295161331/2500 + 121889/75000) \* 1/arctg(4,358905) = (6008854961819/75000) \* 1/arctg(871781/200000) = **5,95548\*10^7**

**Fifth.**

**A:** (-0,000002) **B:** (4,435311) **C:** (-0,324110) **D:** (0,452110)

(2\*(-0,0000023112) + 0,32411 + 4,4353111/3) / arctg(4,4353111 – 0,45211/2) = (202565861/625000000 + 4435311/30000000) \* 1/arctg(4,2092561) = (6759534041/3750000000) \* 1/arctg(42092561/10000000) = **1,34765**

**Висновки**

При виконанні роботи я навчився працювати з командами Ассемблера для арифметики з плаваючою комою та виконав розрахунки з елементами масивів.