**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.”

Виконав: Перевірив:

Студент ІІ курсу ФІОТ доц. Павлов В. Г.

групи ІМ-22

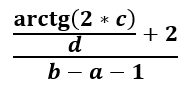
Сачко Максим Євгенійович

номер у списку групи (варіант): 20

Київ 2024

**Мета роботи:** Вивчення прийомів модульного програмування, методів звернення до процедур і передачі в них параметрів.

Завдання за варіантом:

номер у списку групи: 20, тож 20 варіант: **(arctg(2\*c)/d + 2)/(b - a - 1)**, або це можна записати так: .

**1. Вивчити методи звернення до процедур і передачі в них параметрів [1].**

**2. Для свого варіанту індивідуального завдання до лабораторної роботи 6 розробити програму на мові Асемблер, в якій використовувати три процедури з різними способами передачі параметрів:**

* **через регістри;**
* **через стек;**
* **за допомогою директив EXTRN та PUBLIC.**

**3. Для цього чисельник дробу зі свого варіанту індивідуального завдання до лабораторної роботи 6 розділити на два доданка, з яких для першого застосувати передачу параметрів і результату через регістри, а для другого – через стек. Для знаменника використовувати метод оголошення загальних змінних директивами public і extern. Виведення результату\* виконати в основній програмі.**

**\* Вхідні дані та результат у вікні виводу мають бути представлені у десяткової системі счислення.**

**4. Розрахунки (п. 3) повторити в програмі для 5 значень змінних\*\*, причому всі вхідні значення задати дійсними числами у вигляді одновимірних масивів.**

**\*\* Значення змінних a, b, c та d повинні мати бути дійсними числами, не рівними 0 або 1, допустимими для обчислення трансцендентних функцій, а тригонометричні функції обчислюються від аргументів, заданих в радіанах.**

**5. Для перевірки правильності виконання розрахунків і результатів, що виводяться, заздалегідь виконати контрольні розрахунки, які повинні охоплювати різноманітні сполучення вхідних даних, на які програма повинна надавати вірну відповідь\*\*\*. Проміжні і остаточні результати контрольних розрахунків привести в звіті по лабораторній роботі. Точність розрахунків така ж, як і у лаб. роботі 6.**

**\*\*\* Якщо задані значення вхідних змінних не належать до області визначень певних функцій зі завдання, то програма повинна сповіщати про це.**

Моя програма проходить масив з кінця на початок. Тому у коді на змінні треба буде дивитись з кінця масивів. І також змінювати змінні масиву треба з кінця, щоб показувалась відповідь у відповідному порядку. Контрольні розрахунки наведені у тому порядку, у якому будуть виводитись розраховані програмою результати.

**Контрольні розрахунки:**

Варто зазначити, що у формулі використовується функція **arctg**, яка визначена для всієї множини дійсних чисел, а тому додаткова перевірка відповідності аргумента функції її області визначення, а разом із нею й відповідний контрольний розрахунок, де цю область визначення порушено, НЕ ПОТРІБНІ.

Але контрольних розрахунків буде шість. Додатковий контрольний розрахунок буде показувати, що має робити програма, якщо хоч одна зі змінних дорівнює 0 або 1.

1. Чисельник і знаменник більші за нуль.

a = 19.86, b = 21.77, c = 311.87, d = 15.44

(arctg(2\*c)/d + 2)/(b - a - 1) = (arctg(2\*311.87)/15.44 + 2)/(21.77 - 19.86 - 1)

(b - a - 1):

b - a = 21.77 - 19.86 = 1.91

b - a -1 = 1.91 - 1 = 0.91

(arctg(2\*c)/d + 2):

2\*c = 2\*311.87 = 623.74

arctg(2\*c) = arctg(623.74) ≈ 1.56919309605

arctg(2\*c)/d = 1.56919309605/15.44 ≈ 0.10163167721

arctg(2\*c)/d + 2 = 0.10163167721 + 2 ≈ 2.10163167721

(arctg(2\*c)/d + 2)/(b - a - 1) = 2.10163167721/0.91 ≈ 2.30948535957

Таким чином:

Результат = 2.30948535957

2. Одна зі змінних дорівнює 1 (або 0).

a = 139.8, b = 47.85, c = -164.31, d = 0.0

Так як змінна дорівнює нулю, що за умовою лабораторної роботи не має бути, то розв’язку не буде.

3. Чисельник та знаменник менші за нуль.

a = 0.49, b = -8.14, c = 8.79, d = -0.233

(arctg(2\*c)/d + 2)/(b - a - 1) = (arctg(2\*8.79)/-0.233 + 2)/(-8.14 - 0.49 - 1)

(b - a - 1):

b - a = -8.14 - 0.49 = -8.63

b - a -1 = -8.63 - 1 = -9.63

(arctg(2\*c)/d + 2):

2\*c = 2\*8.79 = 17.58

arctg(2\*c) = arctg(17.58) ≈ 1.51397473764

arctg(2\*c)/d = 1.51397473764/-0.233 ≈ -6.49774565512

arctg(2\*c)/d + 2 = -6.49774565512 + 2 ≈ -4.49774565512

(arctg(2\*c)/d + 2)/(b - a - 1) = -4.49774565512/-9.63 ≈ 0.46705562359

Таким чином:

Результат = 0.46705562359

4. Знаменник дорівнює нулю.

a = -311.55, b = -310.55, c = 728.24, d = -11.23

(arctg(2\*c)/d + 2)/(b - a - 1) = (arctg(2\*728.24)/-11.23 + 2)/(-310.55 - -311.55 - 1)

(b - a - 1):

b - a = -310.55 - -311.55 = 1

b - a -1 = 1 - 1 = 0

Так як знаменник дорівнює нулю, далі розв’язку не буде, бо це помилка ділити на нуль.

5. Чисельник більший за нуль, а знаменник менший за нуль.

a = 223.79, b = 223.26, c = -4.91, d = -3.468

(arctg(2\*c)/d + 2)/(b - a - 1) = (arctg(2\*-4.91)/-3.468 + 2)/(223.26 - 223.79 - 1)

(b - a - 1):

b - a = 223.26 - 223.79 = -0.53

b - a -1 = -0.53 - 1 = -1.53

(arctg(2\*c)/d + 2):

2\*c = 2\*-4.91 = -9.82

arctg(2\*c) = arctg(-9.82) ≈ -1.46931316017

arctg(2\*c)/d = -1.46931316017/-3.468 ≈ 0.42367738182

arctg(2\*c)/d + 2 = 0.42367738182 + 2 ≈ 2.42367738182

(arctg(2\*c)/d + 2)/(b - a - 1) = 2.42367738182/-1.53 ≈ -1.58410286394

Таким чином:

Результат = -1.58410286394

6. Чисельник менший за нуль, а знаменник більший за нуль.

a = 782.43, b = 783.49, c = -57.42, d = 0.39

(arctg(2\*c)/d + 2)/(b - a - 1) = (arctg(2\*-57.42)/0.39 + 2)/(783.49 - 782.43 - 1)

(b - a - 1):

b - a = 783.49 - 782.43 = 1.06

b - a -1 = 1.06 - 1 = 0.06

(arctg(2\*c)/d + 2):

2\*c = 2\*-57.42 = -114.84

arctg(2\*c) = arctg(-114.84) ≈ -1.56208877955

arctg(2\*c)/d = -1.56208877955/0.39 ≈ -4.00535584499

arctg(2\*c)/d + 2 = -4.00535584499 + 2 ≈ -2.00535584499

(arctg(2\*c)/d + 2)/(b - a - 1) = -2.00535584499/0.06 ≈ -33.42259741649

Таким чином:

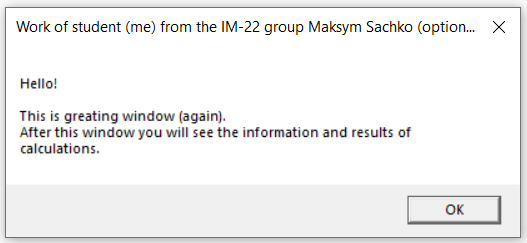
Результат = -33.42259741649

**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.**

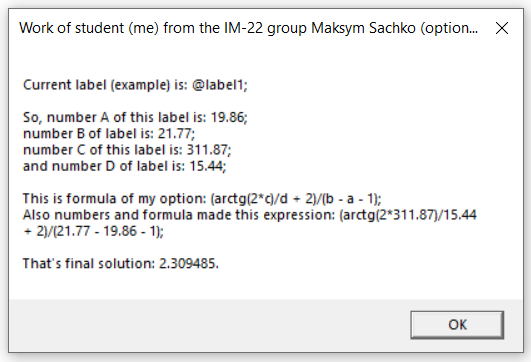
Лістинг програми буде наведено у наступному завданні.

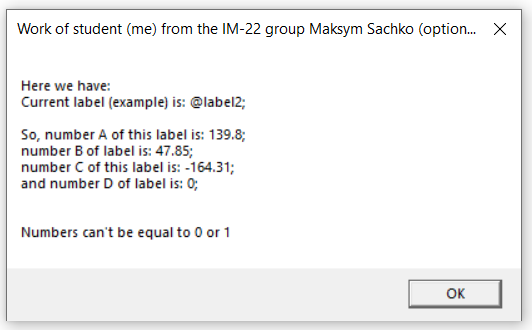
**Скріншоти виконання програми:**

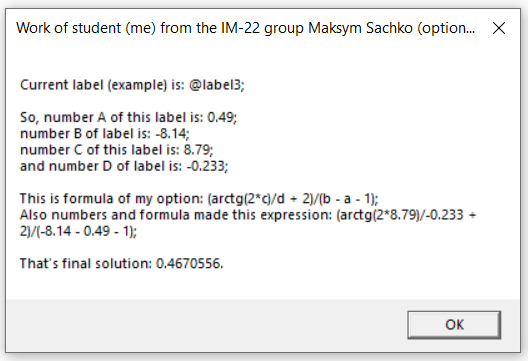
Вікно при запуску програми (прев’ю)

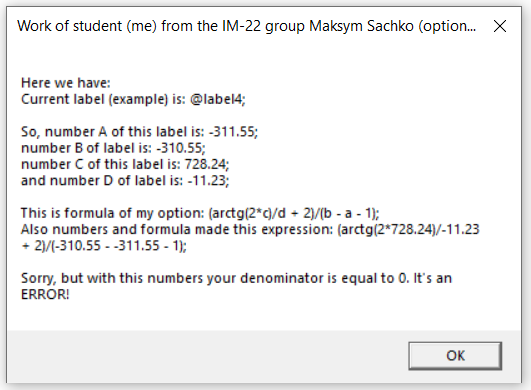


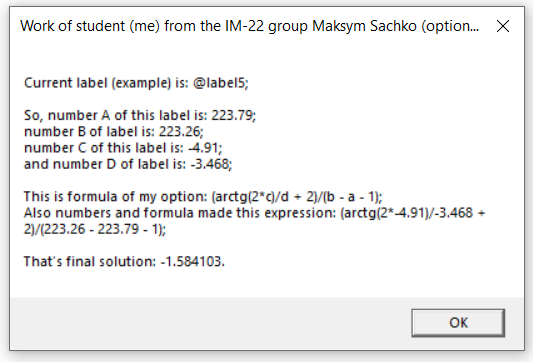
Вікна, які виводять результати розрахунку

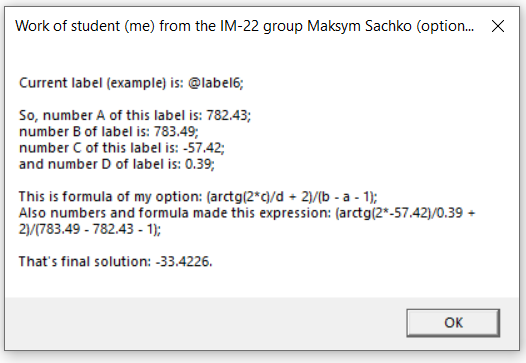












В результаті порівняння контрольних розрахунків з результатами виконання програми видно, що дані збігаються, однак не повністю, а з деякою точністю, так як при контрольних розрахунках виконано округлення до 11 знаків після коми, а під час виконання програми комп’ютер здійснює округлення до 4-7 знаків після коми.

Отже програма працює коректно в усіх сценаріях:

* чисельник та знаменник більше за 0;
* чисельник та знаменник менше за 0;
* чисельник більше за 0, а знаменник – менше;
* чисельник менше за 0, а знаменник – більше;
* випадок, коли знаменник дорівнює 0, опрацьовується коректно (виводиться повідомлення з помилкою). Тут варто також зауважити, що в цьому випадку не відбувається блокування розрахунків для наступних прикладів (якщо в них знаменник не дорівнює нулю), тож у них результат виконання програми коректний.
* і також той випадок, коли хоча б одна зі змінних дорівнює 0 або 1 (виводиться повідомлення з помилкою). Тут також не відбувається блокування розрахунків для наступних прикладів.

**7. У протоколі по лабораторній роботі для першого і другого способів передачі параметрів поруч з відповідними командами у лістингу відобразити в графічному вигляді стани стека при зверненні до процедур, виконання у них команд та повернення з процедур до основної програми.**

**Лістинг програми:**

**(Таблиці йдуть після відповідних операцій. Також операції зі стеком виділено синім кольором, у коментарях позначено, таблиці якого стану стеку відповідає результат виконання цієї операції)**

7-20-IM-22-Sachko.asm:

.386

.model flat, stdcall

option casemap :none

include \masm32\include\masm32rt.inc

include \masm32\macros\macros.asm

public SachkoPolyhymnia\_NumberAforOtherExtern, SachkoCalliope\_NumberBforOtherExtern, SachkoIapetus\_OneChecker

extern SachkoTheseus\_ThisDenominatorCalculation:proto

SachkoHermes\_MessengeWindow macro SachkoIo\_WindowTextHandler, SachkoEuropa\_TitleOfTheWindowHandler

;; So it must be hidden. Frof averyone. Mortals, gods, monsters, etc. Because it is hidden comment

;; invoke wsprintf, addr SachkoEmpusa\_NUmbeRSStrinG8, addr SachkoSisyphus\_nUmberNEed

invoke MessageBox, 0, addr SachkoIo\_WindowTextHandler, addr SachkoEuropa\_TitleOfTheWindowHandler, 0 ;Cottus

;; invoke wsprintf, addr SachkoEmpusa\_NUmbeRSStrinG8, addr SachkoSisyphus\_nUmberNEed

;; It's nice to use normal and hidden comments/. Hooray!!!!!!!!! HOORAY!

endm

SachkoCronus\_DivisionerNumber macro SachkoEchidna\_Divided, SachkoTyphon\_Divisor

fld SachkoEchidna\_Divided

fld SachkoTyphon\_Divisor

;; Dividing number1 / number2 = result

fdiv

fstp SachkoEpimetheus\_TemporalResultBuffer

endm

SachkoAeolus\_LoopingCalculation macro

mov edi, 5

;; This cycle goes through the matrixes in reverse order

mov SachkoMelpomene\_CurrentLabelCeaperBuffer, 0

.repeat

SachkoNotus\_EternalCalculation:

cmp edi, 0

jl SachkoZephyrus\_EndOfEternalLoop

finit

;; First I want to check if numbers A, B, C, D are not equal to 0 or 1

add SachkoMelpomene\_CurrentLabelCeaperBuffer, 1

;; Is A = 0?

fld SachkoPolyhymnia\_AnumbersArray[edi \* 8]

fcom SachkoCrius\_ZeroChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

;; Is A = 1?

fcom SachkoIapetus\_OneChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

fstp SachkoPolyhymnia\_NumberAforOtherExtern

;; Is B = 0?

fld SachkoCalliope\_BnumbersArray[edi \* 8]

fcom SachkoCrius\_ZeroChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

;; Is B = 1?

fcom SachkoIapetus\_OneChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

fstp SachkoCalliope\_NumberBforOtherExtern

;; Is C = 0?

fld SachkoClio\_CnumbersArray[edi \* 8]

fcom SachkoCrius\_ZeroChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

;; Is C = 1?

fcom SachkoIapetus\_OneChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

finit

;; Is D = 0?

fld SachkoEuterpe\_DnumbersArray[edi \* 8]

fcom SachkoCrius\_ZeroChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

;; Is D = 1?

fcom SachkoIapetus\_OneChecker

fstsw ax

sahf

je SachkoEurus\_NotGoodNumbersAnswer

finit

;; After this we can do calculations

;; this is b - a - 1 (I added this comment to not forget what I did and what is left)

call SachkoTheseus\_ThisDenominatorCalculation ;My state of stack #1

|  |  |
| --- | --- |
| My state of stack #1 – call SachkoTheseus\_ThisDenominatorCalculation | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

;; I calculated denomination first to know is it equal to 0 or not. If it is than program is not gonna do a lot of useless calculations (I added this comment to not forget what I did and what is left)

fstp SachkoEos\_DenominatorBuffer

fld SachkoEos\_DenominatorBuffer

fcom SachkoCrius\_ZeroChecker

fstsw ax

sahf

je SachkoCalypso\_ZeroDenominatorAnswer

;; this is for arctg(c\*2)/d + 2

;; this is arctg(2\*c) preparations

lea ebx, SachkoChronos\_NumberTwoForFormula

lea esi, SachkoIapetus\_OneChecker

lea eax, SachkoClio\_CnumbersArray[edi \* 8]

;; this is arctg(c\*2) (I added this comment to not forget what I did and what is left)

call SachkoSyrinx\_MyFirstArgument ;My state of stack #5

|  |  |
| --- | --- |
| My state of stack #5 – call SachkoSyrinx\_MyFirstArgument | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

;; this is arctg(c\*2)/d + 2 preparations

lea edx, SachkoEuterpe\_DnumbersArray[edi \* 8]

push edx ;My state of stack #11

|  |  |
| --- | --- |
| My state of stack #11 – push edx | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
|  |  |
| min |  |

fstp SachkoEpimetheus\_TemporalResultBuffer

push ebx ;My state of stack #12

|  |  |
| --- | --- |
| My state of stack #12 – push ebx | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
|  |  |
| min |  |

lea eax, SachkoEpimetheus\_TemporalResultBuffer

push eax ;My state of stack #13

|  |  |
| --- | --- |
| My state of stack #13 – push eax | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
|  |  |
| min |  |

push esi ;My state of stack #14

|  |  |
| --- | --- |
| My state of stack #14 – push esi | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
|  |  |
| min |  |

;; this is arctg(c\*2)/d + 2(I added this comment to not forget what I did and what is left)

call SachkoParis\_MySecondArgument ;My state of stack #15

|  |  |
| --- | --- |
| My state of stack #15 – call SachkoParis\_MySecondArgument | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

fstp SachkoCoeus\_NumeratorBuffer

;; this is (arctg(c\*2)/d + 2)/(b - a - 1) (I added this comment to not forget what I did and what is left)

SachkoCronus\_DivisionerNumber SachkoCoeus\_NumeratorBuffer, SachkoEos\_DenominatorBuffer

fld SachkoEpimetheus\_TemporalResultBuffer

;; It's final result

fstp SachkoUrania\_ResultBuffer

jmp SachkoBoreas\_ResultAnswer

dec edi

jmp SachkoNotus\_EternalCalculation

SachkoEurus\_NotGoodNumbersAnswer:

invoke wsprintf, addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoMelpomene\_CurrentLabelText,

SachkoMelpomene\_CurrentLabelCeaperBuffer

invoke FloatToStr, SachkoPolyhymnia\_AnumbersArray[edi \* 8], offset SachkoPolyhymnia\_CurrentAbuffer

invoke FloatToStr, SachkoCalliope\_BnumbersArray[edi \* 8], offset SachkoCalliope\_CurrentBbuffer

invoke FloatToStr, SachkoClio\_CnumbersArray[edi \* 8], offset SachkoClio\_CurrentCbuffer

invoke FloatToStr, SachkoEuterpe\_DnumbersArray[edi \* 8], offset SachkoEuterpe\_CurrentDbuffer

;; If one or more numbers equal to 0 (or 1), this window will show

invoke wsprintf, addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoErato\_AllNumbersOfThisOption,

addr SachkoPolyhymnia\_CurrentAbuffer,

addr SachkoCalliope\_CurrentBbuffer,

addr SachkoClio\_CurrentCbuffer,

addr SachkoEuterpe\_CurrentDbuffer

invoke wsprintf, addr SachkoSelene\_NotGoodNumbersTextBuffer,

addr SachkoSelene\_NotGoodNumbersTextWindow,

addr SachkoHyperion\_TextOverall,

addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoHelios\_NotGoodNumbersText

SachkoHermes\_MessengeWindow SachkoSelene\_NotGoodNumbersTextBuffer, SachkoGaia\_GreatingWindowHeader

dec edi

jmp SachkoNotus\_EternalCalculation

SachkoCalypso\_ZeroDenominatorAnswer:

;; If denominator is equal to 0, this window will show

invoke wsprintf, addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoMelpomene\_CurrentLabelText,

SachkoMelpomene\_CurrentLabelCeaperBuffer

invoke FloatToStr, SachkoPolyhymnia\_AnumbersArray[edi \* 8], offset SachkoPolyhymnia\_CurrentAbuffer

invoke FloatToStr, SachkoCalliope\_BnumbersArray[edi \* 8], offset SachkoCalliope\_CurrentBbuffer

invoke FloatToStr, SachkoClio\_CnumbersArray[edi \* 8], offset SachkoClio\_CurrentCbuffer

invoke FloatToStr, SachkoEuterpe\_DnumbersArray[edi \* 8], offset SachkoEuterpe\_CurrentDbuffer

invoke wsprintf, addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoErato\_AllNumbersOfThisOption,

addr SachkoPolyhymnia\_CurrentAbuffer,

addr SachkoCalliope\_CurrentBbuffer,

addr SachkoClio\_CurrentCbuffer,

addr SachkoEuterpe\_CurrentDbuffer

invoke wsprintf, addr SachkoThalia\_FormulaWithNumbersBuffer,

addr SachkoThalia\_FormulaWithNumbers,

addr SachkoClio\_CurrentCbuffer,

addr SachkoEuterpe\_CurrentDbuffer,

addr SachkoCalliope\_CurrentBbuffer,

addr SachkoPolyhymnia\_CurrentAbuffer

invoke wsprintf, addr SachkoTheia\_ZeroDenominatorTextBuffer,

addr SachkoTheia\_ZeroDenominatorTextWindow,

addr SachkoHyperion\_TextOverall,

addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoTerpsichore\_CalculationFormula,

addr SachkoThalia\_FormulaWithNumbersBuffer,

addr SachkoEos\_ZeroDenominatorBad

SachkoHermes\_MessengeWindow SachkoTheia\_ZeroDenominatorTextBuffer, SachkoGaia\_GreatingWindowHeader

dec edi

jmp SachkoNotus\_EternalCalculation

SachkoBoreas\_ResultAnswer:

invoke wsprintf, addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoMelpomene\_CurrentLabelText,

SachkoMelpomene\_CurrentLabelCeaperBuffer

invoke FloatToStr, SachkoPolyhymnia\_AnumbersArray[edi \* 8], offset SachkoPolyhymnia\_CurrentAbuffer

invoke FloatToStr, SachkoCalliope\_BnumbersArray[edi \* 8], offset SachkoCalliope\_CurrentBbuffer

invoke FloatToStr, SachkoClio\_CnumbersArray[edi \* 8], offset SachkoClio\_CurrentCbuffer

invoke FloatToStr, SachkoEuterpe\_DnumbersArray[edi \* 8], offset SachkoEuterpe\_CurrentDbuffer

invoke wsprintf, addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoErato\_AllNumbersOfThisOption,

addr SachkoPolyhymnia\_CurrentAbuffer,

addr SachkoCalliope\_CurrentBbuffer,

addr SachkoClio\_CurrentCbuffer,

addr SachkoEuterpe\_CurrentDbuffer

invoke wsprintf, addr SachkoThalia\_FormulaWithNumbersBuffer,

addr SachkoThalia\_FormulaWithNumbers,

addr SachkoClio\_CurrentCbuffer,

addr SachkoEuterpe\_CurrentDbuffer,

addr SachkoCalliope\_CurrentBbuffer,

addr SachkoPolyhymnia\_CurrentAbuffer

;; Normal window for calculations

invoke FloatToStr, SachkoUrania\_ResultBuffer, offset SachkoUrania\_FinalResultBuffer

;; invoke wsprintf, addr SachkoEmpusa\_NUmbeRSStrinG16, addr SachkoSisyphus\_nUmberNEed

invoke wsprintf, addr SachkoUrania\_ResultTextBuffer,

addr SachkoUrania\_SummaryResultText,

addr SachkoUrania\_FinalResultBuffer

invoke wsprintf, addr SachkoMuses\_FinalCalculationTextBuffer,

addr SachkoMuses\_FinalTextForCalculation,

addr SachkoMelpomene\_CurrentLabelTextBuffer,

addr SachkoErato\_AllCurrentNumbersBuffer,

addr SachkoTerpsichore\_CalculationFormula,

addr SachkoThalia\_FormulaWithNumbersBuffer,

addr SachkoUrania\_ResultTextBuffer

SachkoHermes\_MessengeWindow SachkoMuses\_FinalCalculationTextBuffer, SachkoGaia\_GreatingWindowHeader

dec edi

jmp SachkoNotus\_EternalCalculation

SachkoZephyrus\_EndOfEternalLoop:

invoke ExitProcess, 0

.until edi > -1

endm

.data?

SachkoMuses\_FinalCalculationTextBuffer db 1020 dup(?)

SachkoUrania\_ResultBuffer dq ?

SachkoTheia\_ZeroDenominatorTextBuffer db 811 dup(?)

SachkoCalliope\_CurrentBbuffer db 960 dup(?)

SachkoClio\_CurrentCbuffer db 960 dup(?)

SachkoPolyhymnia\_CurrentAbuffer db 960 dup(?)

SachkoEuterpe\_CurrentDbuffer db 960 dup(?)

SachkoUrania\_FinalResultBuffer db 960 dup(?)

SachkoEos\_DenominatorBuffer dt ?

SachkoRhea\_ArctgBuffer dt ?

SachkoCoeus\_NumeratorBuffer dt ?

SachkoErato\_AllCurrentNumbersBuffer db 811 dup(?)

SachkoSelene\_NotGoodNumbersTextBuffer db 811 dup(?)

SachkoEpimetheus\_TemporalResultBuffer dt ?

SachkoMelpomene\_CurrentLabelTextBuffer db 316 dup(?)

SachkoMelpomene\_CurrentLabelCeaperBuffer dd ?

SachkoThalia\_FormulaWithNumbersBuffer db 811 dup(?)

SachkoUrania\_ResultTextBuffer db 811 dup(?)

SachkoCalliope\_NumberBforOtherExtern dq ?

SachkoPolyhymnia\_NumberAforOtherExtern dq ?

.data

SachkoNumber\_striNG db "%d",0

SachkoCrius\_ZeroChecker dq 0.0

SachkoIapetus\_OneChecker dq 1.0

SachkoCerberus\_truEWronGVERYvroNGpasSwordText db "You wrote something!",10,10,

"Please, leave!",0

SachkoCalliope\_BnumbersArray dq 783.49, 223.26, -310.55, -8.14, 47.85, 21.77

;I just wanted to add this

SachkoSisyphus\_MaGIcWoRdS db "%s",13,10

db "This too!",0

Sachkoerus\_SecOnDseNTfoRerRORwinDow db "Now gone!",0

SachkoHelios\_NotGoodNumbersText db "Numbers can't be equal to 0 or 1",0

SachkoThanatos\_FiRStWInDoWGreAtIng db "%s",10,10,

"%s",10,10,

"%s",10,0

SachkoMelinoe\_OPenAFTERallInforMATIOnTexT db "%s",10,10,

"%s",10,10,

"%s",10,10,

10,10,

"%s",10,10,

"&s",10,10,

"%s",0

SachkoClio\_CnumbersArray dq -57.42, -4.91, 728.24, 8.79, -164.31, 311.87

SachkoOceanus\_NuMbErForSOMthinG dd 18

SachkoCharon\_SecondDDWinDOwGREating db "%s",10,10,

"%s",10,10,

"%s",10,0

SachkoCerberus\_WronGVERYvroNGpasSwordText db "%s",10,10,

"%s",0

SachkoMinthe\_FirsTStringForFIRRstWindow db "Hello!",0

SachkoOrm\_wRongVeryWROngPAasSSworDHeADer db "WRONG, very wrong PASSWORD",0

SachkoTantalus\_NumberFORsOmFIng dd 16

SachkoSelene\_NotGoodNumbersTextWindow db "%s",13,10,

"%s",10,10,

"%s",10,10,

"%s",13,0

SachkoMuses\_FinalTextForCalculation db "%s",10,10,

"%s",10,

"%s",13,

"%s",10,10,

"%s",10,0

SachkoEos\_ZeroDenominatorBad db "Sorry, but with this numbers your denominator is equal to 0. It's an ERROR!",0

SachkoDolos\_thirDsTriNgforFirstWIndow db "Do you want to continue?",0

SachkoDanaussDT\_MagicSenteNCEforALL db "This is useless.",0

SachkoGaia\_GreatingWindowHeader db "Work of student (me) from the IM-22 group Maksym Sachko (option 20)",0

SachkoMoirai\_firstStringForSECondWindow db "I's still a greating window!",0

SachkoCeryneianHind\_numBERBtextFoRm db "%d",0

SachkoGeras\_ThirdstringforSeCoNdWinDoW db "and know what is the end?",0

SachkoPolyhymnia\_AnumbersArray dq 782.43, 223.79, -311.55, 0.49, 139.8, 19.86

SachkoIxion\_STRingFoRme db "%s",0

SachkoAthena\_GreatingWindowText db "Hello!",10,10,

"This is greating window (again).",13,10,

"After this window you will see the information and results of calculations.",10,0

SachkoPhilotes\_seconDstrinGFORsecoNDwiNDOw db "Do you still want to continue",0

SachkoSisyphus\_nUmberNEed dd 52

SachkoTerpsichore\_CalculationFormula db "This is formula of my option: (arctg(2\*c)/d + 2)/(b - a - 1);",0

SachkoLeuce\_AlotOFwordText db "Good day in Underworld!",0

SachkoHyperion\_TextOverall db "Here we have:",0

SachkoTantalus\_nuMBErStriNFgForSomeThiNg db "%d",0

SachkoWhyNot\_Stringforall db "%s",10,10,

"%s",10,10,

"%s",10,0

SachkoErato\_AllNumbersOfThisOption db "So, number A of this label is: %s;",13,

"number B of label is: %s;",13,

"number C of this label is: %s;",13,

"and number D of label is: %s;",10,0

SachkoMelpomene\_CurrentLabelText db "Current label (example) is: @label%d;",0

SachkoThalia\_FormulaWithNumbers db "Also numbers and formula made this expression: (arctg(2\*%s)/%s + 2)/(%s - %s - 1);",0

SachkoUrania\_SummaryResultText db "That's final solution: %s.",0

SachkoTheia\_ZeroDenominatorTextWindow db "%s",13,10,

"%s",10,10,

"%s",10,

"%s",13,

"%s",10,10,

"%s",10,0

SachkoEuterpe\_DnumbersArray dq 0.39, -3.468, -11.23, -0.233, 0.0, 15.44

SachkoChronos\_NumberTwoForFormula dq 2.0

SachkoEmpusa\_NUmbeRSStrinGone db "%d",0

SachkoEmpusa\_NUmbeRSStrinGsecond db "%d",0

SachkoEmpusa\_NUmbeRSStrinGthree db "%d",0

SachkoEmpusa\_NUmbeRSStrinGfourth db "%d",0

SachkoEmpusa\_NUmbeRSStrinGfive db "%d",0

SachkoEmpusa\_NUmbeRSStrinGsixth db "%d",0

SachkoEmpusa\_NUmbeRSStrinG7 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG8 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG9 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG10 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG11 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG12 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG13 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG14 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG15 db "%d",0

SachkoEmpusa\_NUmbeRSStrinG16 db "%d",0

.const

.code

SachkoSyrinx\_MyFirstArgument proc

;; this is c\*2 (I added this comment to not forget what I did and what is left)

call SachkoIris\_MultyplierNumber ;My state of stack #6

|  |  |
| --- | --- |
| My state of stack #6 – call SachkoIris\_MultyplierNumber | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EIP*** | ***Адреса повернення в процедуру SachkoSyrinx\_MyFirstArgument*** |
|  |  |
| min |  |

;; this is arctg(c\*2) (I added this comment to not forget what I did and what is left)

call SachkoHelen\_ArctangensNumber ;My state of stack #8

|  |  |
| --- | --- |
| My state of stack #8 – call SachkoHelen\_ArctangensNumber | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EIP*** | ***Адреса повернення в процедуру SachkoSyrinx\_MyFirstArgument*** |
|  |  |
| min |  |

;; No additional checks needed because you can use every number in arctg

ret ;My state of stack #10

|  |  |
| --- | --- |
| My state of stack #10 – командою ret у кінці виконання процедури SachkoSyrinx\_MyFirstArgument було витягнуто зі стеку адресу повернення з процедури в основну програму | |
| max |  |
| … | … |
|  |  |
|  |  |
| min |  |

SachkoSyrinx\_MyFirstArgument endp

SachkoParis\_MySecondArgument proc

;; So I choosed ebp because it was used on lecture example

push ebp ;My state of stack #16

|  |  |
| --- | --- |
| My state of stack #16 – push ebp | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EBP*** | ***Поточне значення регістра EBP*** |
|  |  |
| min |  |

mov ebp, esp

mov eax, [ebp+16]

mov ecx, [ebp+8]

mov edx, [ebp+12]

mov ebx, [ebp+20]

;; this is arctg(c\*2)/d (I added this comment to not forget what I did and what is left)

call SachkoAchilles\_Division ;My state of stack #17

|  |  |
| --- | --- |
| My state of stack #17 – call SachkoAchilles\_Division | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EBP*** | ***Поточне значення регістра EBP*** |
| ***EIP*** | ***Адреса повернення в процедуру SachkoParis\_MySecondArgument*** |
|  |  |
| min |  |

;; this is arctg(c\*2)/d + 2 (I added this comment to not forget what I did and what is left)

call SachkoAgamemnon\_Additioner ;My state of stack #19

|  |  |
| --- | --- |
| My state of stack #19 – call SachkoAgamemnon\_Additioner | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EBP*** | ***Поточне значення регістра EBP*** |
| ***EIP*** | ***Адреса повернення в процедуру SachkoParis\_MySecondArgument*** |
|  |  |
| min |  |

;; ebp cool

pop ebp ;My state of stack #21

|  |  |
| --- | --- |
| My state of stack #21 – pop ebp – наприкінці виконання процедури SachkoParis\_MySecondArgument повертаємо зі стеку значення EBP | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

;; So now I need to return now

ret 16 ;My state of stack #22

|  |  |
| --- | --- |
| My state of stack #22 – командою ret 16 у кінці виконання процедури SachkoParis\_MySecondArgument було витягнено зі стеку адресу повернення з процедури в основну програму, а також зсунуто стек на 16 байтів, завдяки чому його було повернено в початковий стан | |
| max |  |
| … | … |
|  |  |
|  |  |
| min |  |

SachkoParis\_MySecondArgument endp

SachkoIris\_MultyplierNumber proc

;; Multiplying number1 \* number2 = result

;; 2\*c (I added this comment to not forget what I did and what is left)

fld qword ptr [eax]

fld qword ptr [ebx]

fmul

ret ;My state of stack #7

|  |  |
| --- | --- |
| My state of stack #7 – командою ret у кінці виконання процедури SachkoIris\_MultyplierNumber було витягнуто зі стеку адресу повернення з процедури в процедуру SachkoSyrinx\_MyFirstArgument | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

SachkoIris\_MultyplierNumber endp

SachkoHelen\_ArctangensNumber proc

;; Arctg this arctg(number) = result

;; arctg(temp\_res) or arctg(2\*c) (I added this comment to not forget what I did and what is left)

fld qword ptr [esi]

fpatan

;; No additional checks needed because you can use every number in arctg

ret ;My state of stack #9

|  |  |
| --- | --- |
| My state of stack #9 – командою ret у кінці виконання процедури SachkoHelen\_ArctangensNumber було витягнуто зі стеку адресу повернення з процедури в процедуру SachkoSyrinx\_MyFirstArgument | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

SachkoHelen\_ArctangensNumber endp

SachkoAchilles\_Division proc

;; No additional checks are needed here because d cannot be equal to 0 according to the task of the laboratory work

;; temp\_res/d or arctg(2\*c)/d (I added this comment to not forget what I did and what is left)

fld tbyte ptr [edx]

fld qword ptr [ebx]

fdiv

ret ;My state of stack #18

|  |  |
| --- | --- |
| My state of stack #18 – командою ret у кінці виконання процедури SachkoAchilles\_Division було витягнуто зі стеку адресу повернення з процедури в процедуру SachkoParis\_MySecondArgument | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EBP*** | ***Поточне значення регістра EBP*** |
|  |  |
| min |  |

SachkoAchilles\_Division endp

SachkoAgamemnon\_Additioner proc

;; temp\_res + 2 or arctg(2\*c)/d + 2 (I added this comment to not forget what I did and what is left)

fld qword ptr [eax]

fadd

ret ;My state of stack #20

|  |  |
| --- | --- |
| My state of stack #20 – командою ret у кінці виконання процедури SachkoAgamemnon\_Additioner було витягнуто зі стеку адресу повернення з процедури в процедуру SachkoParis\_MySecondArgument | |
| max |  |
| … | … |
| ***EDX*** | ***Адреса елементу в масиві SachkoEuterpe\_DnumbersArray[edi \* 8]*** |
| ***EBX*** | ***Адреса константи дійсного числа SachkoChronos\_NumberTwoForFormula*** |
| ***EAX*** | ***Адреса проміжного результату SachkoEpimetheus\_TemporalResultBuffer*** |
| ***ESI*** | ***Адреса константи дійсного числа SachkoIapetus\_OneChecker*** |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EBP*** | ***Поточне значення регістра EBP*** |
|  |  |
| min |  |

SachkoAgamemnon\_Additioner endp

SachkoSeventhLabSomeDeities:

invoke MessageBox, 0, addr SachkoAthena\_GreatingWindowText, addr SachkoGaia\_GreatingWindowHeader, 0

SachkoAeolus\_LoopingCalculation

end SachkoSeventhLabSomeDeities

7-20-IM-22-Sachko-denominator.asm:

.386

.model flat, stdcall

option casemap :none

public SachkoTheseus\_ThisDenominatorCalculation

extern SachkoPolyhymnia\_NumberAforOtherExtern:qword, SachkoCalliope\_NumberBforOtherExtern:qword, SachkoIapetus\_OneChecker:qword

.code

SachkoTheseus\_ThisDenominatorCalculation proc

;; this is b - a (I added this comment to not forget what I did and what is left)

fld SachkoCalliope\_NumberBforOtherExtern

fld SachkoPolyhymnia\_NumberAforOtherExtern

fsub

;; this is b - a - 1 (I added this comment to not forget what I did and what is left)

call SachkoClytie\_SubtractionerNumber ;My state of stack #2

|  |  |
| --- | --- |
| My state of stack #2 – call SachkoClytie\_SubtractionerNumber | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
| ***EIP*** | ***Адреса повернення в процедуру SachkoTheseus\_ThisDenominatorCalculation*** |
|  |  |
| min |  |

ret ;My state of stack #4

|  |  |
| --- | --- |
| My state of stack #4 – командою ret у кінці виконання процедури SachkoTheseus\_ThisDenominatorCalculation було витягнуто зі стеку адресу повернення з процедури в основну програму | |
| max |  |
| … | … |
|  |  |
|  |  |
| min |  |

SachkoTheseus\_ThisDenominatorCalculation endp

SachkoClytie\_SubtractionerNumber proc

;; this is b - a - 1 or temp\_res - 1

fld SachkoIapetus\_OneChecker

fsub

ret ;My state of stack #3

|  |  |
| --- | --- |
| My state of stack #3 – командою ret у кінці виконання процедури SachkoClytie\_SubtractionerNumber було витягнуто зі стеку адресу повернення з процедури в процедуру SachkoTheseus\_ThisDenominatorCalculation | |
| max |  |
| … | … |
| ***EIP*** | ***Адреса повернення в основну програму*** |
|  |  |
| min |  |

SachkoClytie\_SubtractionerNumber endp

end

7-20-IM-22-Sachko.bat:

@echo off

ml /c /coff "7-20-IM-22-Sachko.asm"

ml /c /coff "7-20-IM-22-Sachko-denominator.asm"

link /subsystem:windows "7-20-IM-22-Sachko.obj" "7-20-IM-22-Sachko-denominator.obj"

7-20-IM-22-Sachko.exe

**8. Зробити висновки по лабораторній роботі.**

**Висновок:** У результаті порівняння контрольних розрахунків із результатами виконання програми для всіх передбачених лабораторною роботою сценаріїв (перевірка аргумента функції на відповідність її області визначення, а також відповідний приклад, у якому цю область визначення порушено, реалізовано не було, оскільки в моєму варіанті використовується функція arctg, яка визначена для всієї множини дійсних чисел, а тому такої перевірки не потребує):

* чисельник та знаменник більше за 0;
* чисельник та знаменник менше за 0;
* чисельник більше за 0, а знаменник – менше;
* чисельник менше за 0, а знаменник – більше;
* знаменник дорівнює 0;
* хоча б одна зі змінних дорівнює 0 або 1,

бачимо, що результати обчислень співпадають з деякою точністю (похибка виникає через те, що в ході виконання програми округлення здійснювалось із меншою точністю, ніж у контрольних розрахунках), і випадки, коли знаменник дорівнює 0, і коли одна зі змінних дорівнює нулю або одиниці, обробляються коректно (і після цих випадків не блокується виконання розрахунків для наступних прикладів, якщо вони відповідають умові, що знаменник не дорівнює нулю і змінні не дорівнюють нулю або одиниці).

У ході виконання роботи програму з лабораторної роботи №6 було перероблено так, що процес обчислення виразу було розбито на три процедури, для кожної з яких використовується свій спосіб передачі параметрів (через регістри, через стек, за допомогою директив extern та public). Враховуючи результати порівняння виконання програми та контрольних розрахунків, можна зробити висновок, що всі три процедури функціонують коректно. Також було досліджено стани стеку під час виконання програми й виклику процедур. При передачі параметрів у процедуру через стек, всередині процедури вони доступні за допомогою звернення до регістра ebp (він же base pointer), який встановлюється в початку виконання процедури та значення якого рівне значенню регістра esp (він же stack pointer). Також при виклику кожної процедури в стек додається значення регістра EIP, яке являє собою адресу повернення в основну програму (або процедуру, у якій викликали іншу процедуру) з процедури. При виході з процедури за допомогою команди ret ця адреса повертається зі стеку для подальшого повернення в основну програму.

Таким чином, роблю висновок, що програму написано коректно.

У ході виконання цієї лабораторної роботи мною було вивчено прийоми модульного програмування, а також методи звернення до процедур та передачі в них параметрів (через регістри, стек або за допомогою директив public та extern).