Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. 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»

Виконав: Перевірив:

Студент 2 курсу ФІОТ доц. Павлов В.Г.

групи ІМ-33

Родіна Олександр Вікторович

номер у списку групи - 19

Київ 2025

**Мета роботи:**

Вивчення команд Асемблера для арифметики з плаваючою

комою і здобуття навичок виконання розрахунків з елементами ма-

сивів.

**Порядок виконання роботи**

1. Вивчити арифметичні команди з плаваючою комою [1].

2. Розробити програму на мові Асемблер, в якій згідно з ін-

дивідуальним варіантом завдання (табл. 1) виконуються обчислен-

ня значення арифметичного вираження із застосуванням команд

співпроцесора ix87 з подальшим виведенням результату\* у вікон-

ному інтерфейсі.

3. Для всіх варіантів передбачити завдання значень вхідних

змінних у форматі double (DQ), проміжних результатів обчислень –

у форматі long double (DT), а кінцевих - знову у форматі double.

4. Розрахунки (п. 2, 3) повторити в програмі для 5 значень

змінних\*\*, причому всі вхідні значення задати дійсними числами у

вигляді одновимірних масивів.

5. Для перевірки правильності виконання розрахунків і ре-

зультатів, що виводяться, заздалегідь виконати контрольні розра-

хунки. Проміжні і остаточні результати контрольних розрахунків з

точністю не гірше точності розрахунків у комп'ютеру привести в

звіті.

6. Виконати відладку програми шляхом порівняння розрахо-

ваних 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.

**\* Вхідні дані та результат у вікні виводу мають бути предста-**

**влені у десяткової системі счислення.**

**\*\* Значення змінних a, b, c та d повинні мати бути дійсними**

**числами, не рівними 0 або 1, допустимими для обчислення тра-**

**нсцендентних функцій, а тригонометричні функції обчислю-**

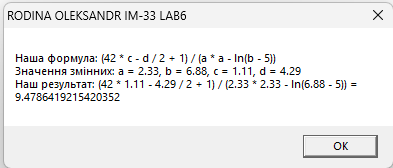
**ються від аргументів, заданих в радіанах.**

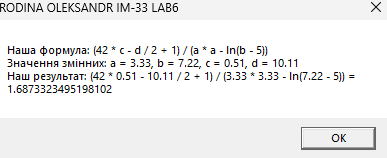
**Хід виконання роботи**

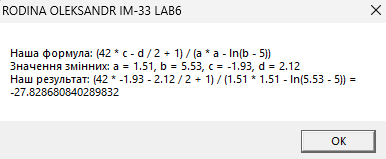
**Вибір потрібної формули згідно варіанту (пункт 2)**Мій варіант згідно списку групи: 19

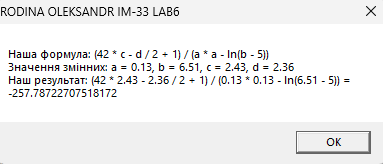
Формула по варіанту:  
  

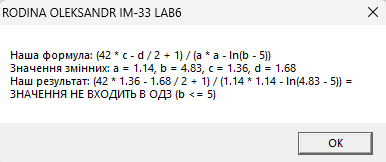

**Проведення розрахунків за допомогою програми (пункт 4)**

****

****

****

****

****

**Виконання контрольних розрахунків (пункт 5)**

**Випадок 1:**

* **a = 2.33**
* **b = 6.88**
* **c = 1.11**
* **d = 4.29**

***Обчислення:***

1. **Чисельник:**

**42.00 \* c = 42.00 \* 1.11 = 46.6200**

**d / 2.00 = 4.29 / 2.00 = 2.1450**

**42 \* c - d / 2 = 46.6200 - 2.1450 = 44.4750**

**42 \* c - d / 2 + 1.00 = 44.4750 + 1.00 = 45.4750**

**Чисельник = 45.4750**

1. **Знаменник:**

**a \* a = 2.33 \* 2.33 = 5.4289**

**b - 5.00 = 6.88 - 5.00 = 1.88**

**ln(b - 5) = ln(1.88) ≈ 0.631278**

**a \* a - ln(b - 5) = 5.4289 - 0.631278 = 4.797622**

**Знаменник = 4.797622**

1. **Результат:**

**Чисельник / Знаменник = 45.4750 / 4.797622**

**45.4750 / 4.797622 ≈ 9.47864**

**Випадок 2:**

* **a = 3.33**
* **b = 7.22**
* **c = 0.51**
* **d = 10.11**

***Обчислення:***

1. **Чисельник:**

**42.00 \* c = 42.00 \* 0.51 = 21.4200**

**d / 2.00 = 10.11 / 2.00 = 5.0550**

**42 \* c - d / 2 = 21.4200 - 5.0550 = 16.3650**

**42 \* c - d / 2 + 1.00 = 16.3650 + 1.00 = 17.3650**

**Чисельник = 17.3650**

1. **Знаменник:**

**a \* a = 3.33 \* 3.33 = 11.0889**

**b - 5.00 = 7.22 - 5.00 = 2.22**

**ln(b - 5) = ln(2.22) ≈ 0.797507**

**a \* a - ln(b - 5) = 11.0889 - 0.797507 = 10.291393**

**Знаменник = 10.291393**

1. **Результат:**

**Чисельник / Знаменник = 17.3650 / 10.291393**

**17.3650 / 10.291393 ≈ 1.6873**

**Випадок 3:**

* **a = 1.51**
* **b = 5.53**
* **c = -1.93**
* **d = 2.12**

***Обчислення:***

1. **Чисельник:**

**42.00 \* c = 42.00 \* (-1.93) = -81.0600**

**d / 2.00 = 2.12 / 2.00 = 1.0600**

**42 \* c - d / 2 = -81.0600 - 1.0600 = -82.1200**

**42 \* c - d / 2 + 1.00 = -82.1200 + 1.00 = -81.1200**

**Чисельник = -81.1200**

1. **Знаменник:**

**a \* a = 1.51 \* 1.51 = 2.2801**

**b - 5.00 = 5.53 - 5.00 = 0.53**

**ln(b - 5) = ln(0.53) ≈ -0.634878**

**a \* a - ln(b - 5) = 2.2801 - (-0.634878) = 2.2801 + 0.634878 = 2.914978**

**Знаменник = 2.914978**

1. **Результат:**

**Чисельник / Знаменник = -81.1200 / 2.914978**

**-81.1200 / 2.914978 ≈ -27.8286**

**Випадок 4:**

* **a = 0.13**
* **b = 6.51**
* **c = 2.43**
* **d = 2.36**

***Обчислення:***

1. **Чисельник:**

**42.00 \* c = 42.00 \* 2.43 = 102.0600**

**d / 2.00 = 2.36 / 2.00 = 1.1800**

**42 \* c - d / 2 = 102.0600 - 1.1800 = 100.8800**

**42 \* c - d / 2 + 1.00 = 100.8800 + 1.00 = 101.8800**

**Чисельник = 101.8800**

1. **Знаменник:**

**a \* a = 0.13 \* 0.13 = 0.0169**

**b - 5.00 = 6.51 - 5.00 = 1.51**

**ln(b - 5) = ln(1.51) ≈ 0.412110**

**a \* a - ln(b - 5) = 0.0169 - 0.412110 = -0.395210**

**Знаменник = -0.395210**

1. **Результат:**

**Чисельник / Знаменник = 101.8800 / (-0.395210)**

**101.8800 / (-0.395210) ≈ -257.78**

**Випадок 5:**

* **a = 1.14**
* **b = 4.83**
* **c = 1.36**
* **d = 1.68**

***Обчислення:***

1. **Чисельник:**

**42.00 \* c = 42.00 \* 1.36 = 57.1200**

**d / 2.00 = 1.68 / 2.00 = 0.8400**

**42 \* c - d / 2 = 57.1200 - 0.8400 = 56.2800**

**42 \* c - d / 2 + 1.00 = 56.2800 + 1.00 = 57.2800**

**Чисельник = 57.2800**

1. **Знаменник:**

**a \* a = 1.14 \* 1.14 = 1.2996**

**b - 5.00 = 4.83 - 5.00 = -0.17**

**ln(b - 5) = ln(-0.17)**

**Помилка: Натуральний логарифм (ln) не визначений для від'ємних чисел**

**Лістинг програми та порівняння розрахунків (пункт 6)  
  
Лістинг моєї програми**

.386

.model flat, stdcall

option casemap: none

include \masm32\include\masm32rt.inc

include \masm32\include\dialogs.inc

.data?

RodinaLab6Result dq ?

RodinaLab6tmms db 512 dup (?)

RodinaLab6strAA db 32 dup (?)

RodinaLab6Denominator dq ?

RodinaLab6strBB db 32 dup (?)

RodinaLab6FiveMin dq ?

RodinaLab6strDD db 32 dup (?)

RodinaLab6Numerator dq ?

RodinaLab6FiveMinVAAA dq ?

RodinaLab6strCC db 32 dup (?)

RodinaLab6StrResult db 32 dup (?)

.data

RodinaLab6AITER42 db "bbbbb",0 ; 5 ітерацій для 5 наборів даних

; Значення

RodinaLab6AAAAA dq 2.33, 3.33, 1.51, 0.13, 1.14

RodinaLab6BBBBB dq 6.88, 7.22, 5.53, 6.51, 4.83

RodinaLab6DDDDD dq 4.29, 10.11, 2.12, 2.36, 1.68

RodinaLab6CCCCC dq 1.11, 0.51, -1.93, 2.43, 1.36

;Константи

RodinaLab6Onee dq 1.0

RodinaLab6Zero dq 0.0

RodinaLab6Twoo dq 2.0

RodinaLab6FortyTwoo dq 42.0

RodinaLab6Fiveeeee dq 5.0

;Заголовки та шаблони

;Наша формула: (42 \* c - d / 2 + 1) / (a \* a - ln(b - 5))

RodinaOleksandrWOW8093 db "Наша формула: (42 \* c - d / 2 + 1) / (a \* a - ln(b - 5))",13,

"Значення змінних: a = %s, b = %s, c = %s, d = %s",13,

"Наш результат: (42 \* %s - %s / 2 + 1) / (%s \* %s - ln(%s - 5)) = %s",0

RodinaOleksandrOHNO8093 db "Наша формула: (42 \* c - d / 2 + 1) / (a \* a - ln(b - 5))",13,

"Значення змінних: a = %s, b = %s, c = %s, d = %s",13,

"Наш результат: (42 \* %s - %s / 2 + 1) / (%s \* %s - ln(%s - 5)) = ДІЛЕННЯ НА 0",0

RodinaOleksandrOHNOLOG1 db "Наша формула: (42 \* c - d / 2 + 1) / (a \* a - ln(b - 5))",13,

"Значення змінних: a = %s, b = %s, c = %s, d = %s",13,

"Наш результат: (42 \* %s - %s / 2 + 1) / (%s \* %s - ln(%s - 5)) = ЗНАЧЕННЯ НЕ ВХОДИТЬ В ОДЗ (b <= 5)",0

RodinaOleksandrMessageBoxCaption db "RODINA OLEKSANDR IM-33 LAB6",0

RodinaLab6SHOWTHIS macro message, header

invoke MessageBox, 0, addr message, addr header, MB\_OK

endm

.code

start:

mov esi, offset RodinaLab6AITER42

mov edi, 0

RodinaCalculationHOI4:

cmp byte ptr [esi], 0

je RodinaLab6ENDTHISNOW

; Конвертуємо поточні значення в рядки для виводу

invoke FloatToStr2, RodinaLab6AAAAA[edi \* 8], addr RodinaLab6strAA

invoke FloatToStr2, RodinaLab6BBBBB[edi \* 8], addr RodinaLab6strBB

invoke FloatToStr2, RodinaLab6CCCCC[edi \* 8], addr RodinaLab6strCC

invoke FloatToStr2, RodinaLab6DDDDD[edi \* 8], addr RodinaLab6strDD

finit

fld RodinaLab6BBBBB[edi \* 8]

fsub RodinaLab6Fiveeeee

ftst

fstsw ax

sahf

jna RodinaLab6MadeOHNOLOG

fstp RodinaLab6FiveMin

fldln2

fld RodinaLab6FiveMin

fyl2x

fstp RodinaLab6FiveMinVAAA

fld RodinaLab6AAAAA[edi \* 8]

fmul st(0), st(0)

fsub RodinaLab6FiveMinVAAA

fstp RodinaLab6Denominator

fld RodinaLab6Denominator

ftst

fstsw ax

sahf

je RodinaLab6MadeOHNO

fld RodinaLab6FortyTwoo

fmul RodinaLab6CCCCC[edi \* 8]

fld RodinaLab6DDDDD[edi \* 8]

fdiv RodinaLab6Twoo

fsubp st(1), st(0)

fadd RodinaLab6Onee

fstp RodinaLab6Numerator

fld RodinaLab6Numerator

fld RodinaLab6Denominator

fdivp st(1), st(0)

fstp qword ptr [RodinaLab6Result]

RodinaLab6MadeWOW:

invoke FloatToStr2, RodinaLab6Result, addr RodinaLab6StrResult

; Формуємо повідомлення про успіх

invoke wsprintf, addr RodinaLab6tmms, addr RodinaOleksandrWOW8093, \

addr RodinaLab6strAA, addr RodinaLab6strBB, addr RodinaLab6strCC, addr RodinaLab6strDD, \

addr RodinaLab6strCC, addr RodinaLab6strDD,

addr RodinaLab6strAA, addr RodinaLab6strAA, addr RodinaLab6strBB, \

addr RodinaLab6StrResult ; Результат

RodinaLab6SHOWTHIS RodinaLab6tmms, RodinaOleksandrMessageBoxCaption

jmp RodinaLab6IterSuperHoi4

RodinaLab6MadeOHNO:

; Формуємо повідомлення про ділення на 0

invoke wsprintf, addr RodinaLab6tmms, addr RodinaOleksandrOHNO8093, \

addr RodinaLab6strAA, addr RodinaLab6strBB, addr RodinaLab6strCC, addr RodinaLab6strDD, \

addr RodinaLab6strCC, addr RodinaLab6strDD, \

addr RodinaLab6strAA, addr RodinaLab6strAA, addr RodinaLab6strBB

RodinaLab6SHOWTHIS RodinaLab6tmms, RodinaOleksandrMessageBoxCaption

jmp RodinaLab6IterSuperHoi4

RodinaLab6MadeOHNOLOG:

; Формуємо повідомлення про помилку ОДЗ

invoke wsprintf, addr RodinaLab6tmms, addr RodinaOleksandrOHNOLOG1, \

addr RodinaLab6strAA, addr RodinaLab6strBB, addr RodinaLab6strCC, addr RodinaLab6strDD, \

addr RodinaLab6strCC, addr RodinaLab6strDD, \

addr RodinaLab6strAA, addr RodinaLab6strAA, addr RodinaLab6strBB

RodinaLab6SHOWTHIS RodinaLab6tmms, RodinaOleksandrMessageBoxCaption

RodinaLab6IterSuperHoi4:

inc esi

inc edi

jmp RodinaCalculationHOI4

RodinaLab6ENDTHISNOW:

invoke ExitProcess, 0 ; Завершення програми

end start

**Порівняння моїх розрахунків**

| **Мої Змінні** | **Результат програми** | **Результат контрольних розрахунків** | **Значення однакові?** |
| --- | --- | --- | --- |
| a = 2.33, b = 6.88, c = 1.11, d = 4.29 | 9.47864 | 9.47864 | Так |
| a = 3.33, b = 7.22, c = 0.51, d =10.11 | 1.6873 | 1.6873 | Так |
| a = 1.51, b = 5.53, c = -1.93, d = 2.12 | -27.8286 | -27.8286 | Так |
| a = 0.13, b = 6.51, c = 2.43, d = 2.36 | -257.78 | -257.78 | Так |
| a = 1.14, b = 4.83, c = 1.36, d = 1.68 | ПОМИЛКА | ПОМИЛКА | Так |

Як можемо бачити значення контрольних розрахунків співпадають зі значеннями з програмною реалізації. З цього можна зробити висновок про те, що програма працює коректно

**Висновки**

Виконання лабораторної роботи значно поглибило моє розуміння використання масивів та операцій з плаваючою комою в середовищі MASM32. Досвід роботи з цими концепціями виявився для мене новим та потребував ретельного опрацювання кожного етапу. Необхідність вручну перевіряти обчислення на контрольних прикладах дозволила не лише переконатися у правильності написаного коду, але й глибше усвідомити особливості арифметики з плаваючою комою на низькому рівні. Особливу увагу було приділено обробці потенційних помилок, таких як ділення на нуль та передача некоректних аргументів до логарифмічної функції (чисел, менших або рівних нулю). Впровадження відповідних перевірок забезпечило стабільну та передбачувану роботу програми навіть у нестандартних ситуаціях. Результатом виконаної роботи стало повне співпадіння обчислених вручну значень з результатами, отриманими за допомогою програми на MASM32. Це підтверджує не лише коректність розробленого алгоритму та його програмної реалізації, але й мою здатність точно та ефективно застосовувати отримані знання на практиці. Представлена у звіті порівняльна таблиця наочно демонструє цю відповідність, підкреслюючи успішність виконання поставленого завдання та засвоєння ключових принципів роботи з масивами та операціями з плаваючою комою в асемблері MASM32. Цей досвід став цінним кроком у моєму розвитку як програміста, розширивши мої навички в області низькорівневого програмування та обробки числових даних.