**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 ix87”

Виконав: Перевірив:

Студент 2-го курсу групи ІМ-13 Павлов Валерій Георгійович

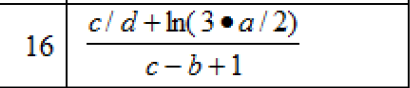
Нестеров Дмитро Васильович  
номер у списку групи: 16

Номер залікової книжки:IM-1320

**Київ 2023**

**Мета роботи**

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

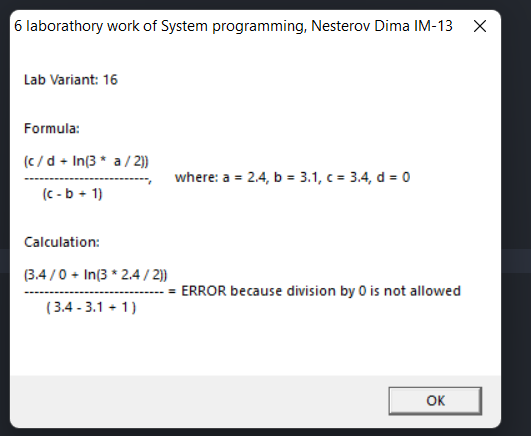


**Скріншоти виконання програми**

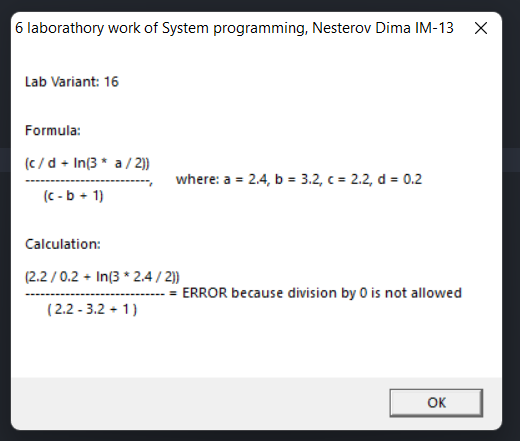
Я зробив 8 прикладів, чотири з яких перевіряють виключні ситуації, в тому числі знаменник дробу c/d, що дорівнює нулю а також нульове а під логарифмом, щоб повністю перевірити правильність реагування програми на числа, що не входять в одз функцій:

* При діленні c/d знаменник d дорівнює нулю
* Знаменник загального дробу c-b+1 дорівнює нулю
* Коли вираз під логарифмом не входить в його область визначення, для логарифму я зробив два випадки:
* коли a = 0
* коли а < 0

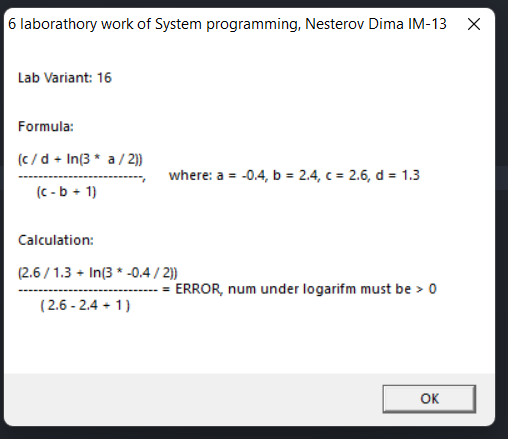
Інші чотири варіанти перевіряють правильність обрахунків для всіх комбінацій знаків чисельника і знаменника головного дробу.



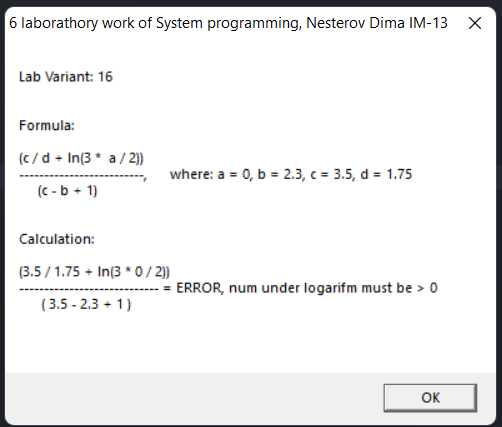
У цьому випадку d дорівнює нулю і при діленні на нього ми маємо виключну ситуацію.



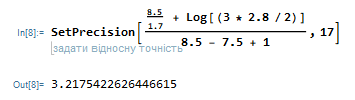
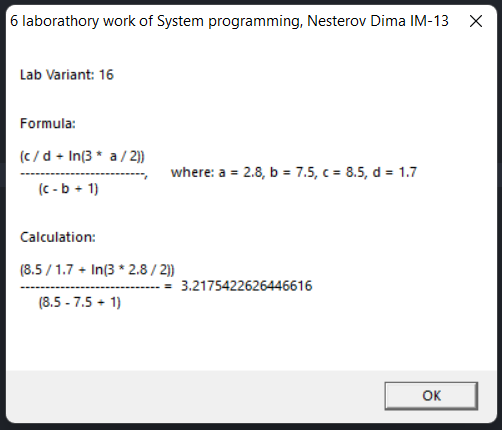
В даному випадку 2.2 – 3.2 + 1 = 0, отже ми знову маємо виключну ситуацію при знаменнику, що дорівнює нулю.



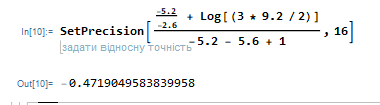
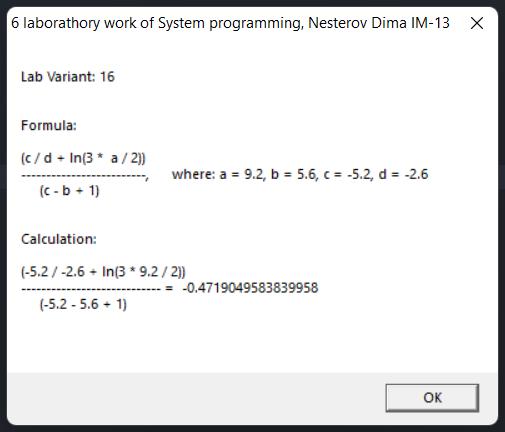
Знову виключна ситуація. У цьому випадку 3 \* (-0.4) / 2 = - 0.6, що менше нуля, що не входить в область визначення логарифму.



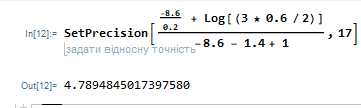
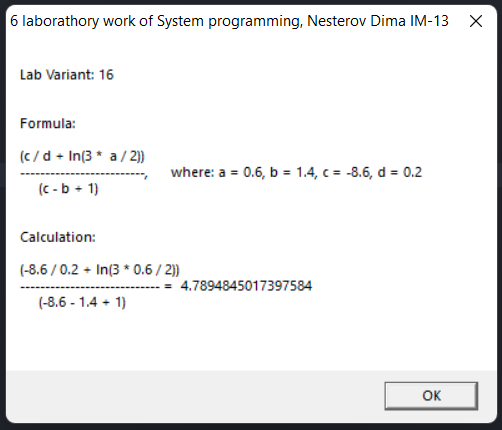
Число, що дорівнює нулю 3 \* 0 / 2 = 0 теж не входить в область визначення.



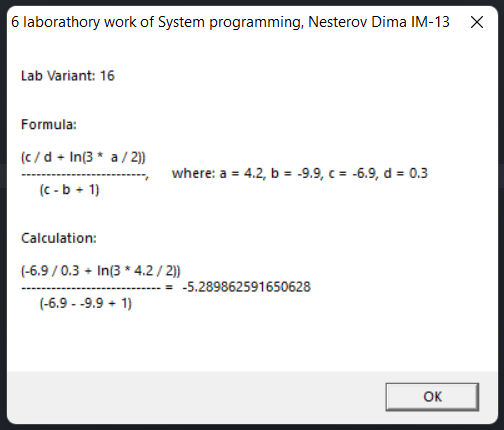
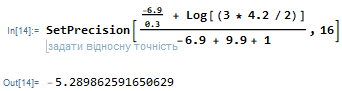
3.2175422626446615 3.2175422626446616, відрізняється лише остання цифра.



-0.4719049583839958 -0.4719049583839958, співпало повністю.



4.7894845017397580 4.7894845017397584, відрізняється лише остання цифра.

- 5.289862591650629 - 5.289862591650628, відрізняється лише остання цифра.

**Лістинг програми**

.386

.MODEL flat, stdcall

OPTION casemap :none

INCLUDE \masm32\include\masm32rt.inc

.DATA

NESTER\_ARRAY\_A dq 2.4, 2.4, -0.4, 0.0, 2.8, 9.2, 0.6, 4.2

NESTER\_ARRAY\_B dq 3.1, 3.2, 2.4, 2.3, 7.5, 5.6, 1.4, -9.9 ;;my arrays with a b с and d =)

NESTER\_ARRAY\_C dq 3.4, 2.2, 2.6, 3.5, 8.5, -5.2, -8.6, -6.9

NESTER\_ARRAY\_D dq 0.0, 0.2, 1.3, 1.75, 1.7, -2.6, 0.2, 0.3

NESTER\_RESULT dq 0.0

NESTER\_CONST3 dq 3.0

NESTER\_CONST2 dq 2.0

NESTER\_CONST1 dq 1.0

NESTER\_TEXT\_ARRAY\_A db 64 dup(?)

NESTER\_TEXT\_ARRAY\_B db 64 dup(?)

NESTER\_TEXT\_ARRAY\_C db 64 dup(?)

NESTER\_TEXT\_ARRAY\_D db 64 dup(?)

NESTER\_TEXT\_RESULT db 32 dup(?)

NESTER\_MESSAGE db 512 dup(?)

NESTER\_WINDOW\_TITLE db "6 laborathory work of System programming, Nesterov Dima IM-13", 0

NESTER\_MESSAGE\_FORMAT db "Lab Variant: 16", 10, 10, 10,

"Formula:", 10, 10, "(c / d + ln(3 \* a / 2))", 10, "-------------------------,", " where: a = %s, b = %s, c = %s, d = %s", 10, " (c - b + 1)", 10, 10, 10,

"Calculation:", 10, 10,

"(%s / %s + ln(3 \* %s / 2))", 10, "---------------------------- = %s", 10, " (%s - %s + 1)", 10, 10, 10,0

NESTER\_ZERO\_DENOMITATOR\_FORMAT db "Lab Variant: 16", 10, 10, 10,

"Formula:", 10, 10, "(c / d + ln(3 \* a / 2))", 10, "-------------------------,", " where: a = %s, b = %s, c = %s, d = %s", 10, " (c - b + 1)", 10, 10, 10,

"Calculation:", 10, 10,

"(%s / %s + ln(3 \* %s / 2))", 10, "---------------------------- = ERROR because division by 0 is not allowed", 10, " ( %s - %s + 1 )", 10, 10, 10, 0

NESTER\_INCORRECT\_AREA\_FORMAT db "Lab Variant: 16", 10, 10, 10,

"Formula:", 10, 10, "(c / d + ln(3 \* a / 2))", 10, "-------------------------,", " where: a = %s, b = %s, c = %s, d = %s", 10, " (c - b + 1)", 10, 10, 10,

"Calculation:", 10, 10,

"(%s / %s + ln(3 \* %s / 2))", 10, "---------------------------- = ERROR, num under logarifm must be > 0", 10, " ( %s - %s + 1 )", 10, 10, 10, 0

.CODE

NESTER\_TO\_STR MACRO

INVOKE FloatToStr2, NESTER\_ARRAY\_A[esi\*8], addr NESTER\_TEXT\_ARRAY\_A[esi\*8]

INVOKE FloatToStr2, NESTER\_ARRAY\_B[esi\*8], addr NESTER\_TEXT\_ARRAY\_B[esi\*8]

INVOKE FloatToStr2, NESTER\_ARRAY\_C[esi\*8], addr NESTER\_TEXT\_ARRAY\_C[esi\*8]

INVOKE FloatToStr2, NESTER\_ARRAY\_D[esi\*8], addr NESTER\_TEXT\_ARRAY\_D[esi\*8]

INVOKE FloatToStr2, NESTER\_RESULT, addr NESTER\_TEXT\_RESULT

ENDM

NESTER\_CREATE\_MESSAGE MACRO

NESTER\_TO\_STR

INVOKE wsprintf, addr NESTER\_MESSAGE, addr NESTER\_MESSAGE\_FORMAT,

addr NESTER\_TEXT\_ARRAY\_A[esi \* 8], addr NESTER\_TEXT\_ARRAY\_B[esi \* 8], addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_D[esi \* 8],

addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_D[esi \* 8], addr NESTER\_TEXT\_ARRAY\_A[esi \* 8],

addr NESTER\_TEXT\_RESULT,addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_B[esi \* 8]

ENDM

NESTER\_CREATE\_ERROR\_MESSAGE MACRO param

NESTER\_TO\_STR

INVOKE wsprintf, addr NESTER\_MESSAGE, param ,

addr NESTER\_TEXT\_ARRAY\_A[esi \* 8], addr NESTER\_TEXT\_ARRAY\_B[esi \* 8], addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_D[esi \* 8],

addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_D[esi \* 8], addr NESTER\_TEXT\_ARRAY\_A[esi \* 8],

addr NESTER\_TEXT\_ARRAY\_C[esi \* 8], addr NESTER\_TEXT\_ARRAY\_B[esi \* 8]

ENDM

NESTER\_CREATE\_ZERO\_DENOMINATOR\_ERROR\_MESSAGE MACRO

NESTER\_CREATE\_ERROR\_MESSAGE addr NESTER\_ZERO\_DENOMITATOR\_FORMAT

ENDM

NESTER\_CREATE\_INVALID\_AREA\_ERROR\_MESSAGE MACRO

NESTER\_CREATE\_ERROR\_MESSAGE addr NESTER\_INCORRECT\_AREA\_FORMAT,

ENDM

NESTER\_CALCULATE MACRO

FINIT

FLD NESTER\_ARRAY\_C[esi\*8] ;c

FLD NESTER\_ARRAY\_D[esi\*8] ; d

FTST

FNSTSW ax

SAHF

JZ NESTER\_ZERO\_DENOMINATOR\_ERROR ; checking for zero denominator for c/d expression

FDIV ; dividing c on d

FLD NESTER\_ARRAY\_A[esi\*8] ;a

FTST

FSTSW AX

SAHF

JBE NESTER\_INCORRECT\_AREA ; checking if a <= 0

FMUL NESTER\_CONST3 ;multiplying 3 and a

FDIV NESTER\_CONST2 ;dividing 3a on 2

FLDln2

FXCH st(1)

FYL2X ;ln(3a/2)

FADD ;adding c/d to ln(3a/2)

FLD NESTER\_ARRAY\_C[esi\*8] ;c

FSUB NESTER\_ARRAY\_B[esi\*8] ; subtractiob b from c

FADD NESTER\_CONST1 ; adding 1 to c-b

FTST

FNSTSW ax

SAHF

JZ NESTER\_ZERO\_DENOMINATOR\_ERROR ;checking if denominator c-b+1 = 0

FDIV ; final dividing

FSTP NESTER\_RESULT ;result will be (ln(3a/2) + c/d)/(c-b+1)

NESTER\_CREATE\_MESSAGE

INVOKE MessageBox, 0, addr NESTER\_MESSAGE, addr NESTER\_WINDOW\_TITLE, 0

JMP NESTER\_SKIP\_ERRORS

NESTER\_ZERO\_DENOMINATOR\_ERROR:

NESTER\_CREATE\_ZERO\_DENOMINATOR\_ERROR\_MESSAGE

INVOKE MessageBox, 0, addr NESTER\_MESSAGE, addr NESTER\_WINDOW\_TITLE, 0

JMP NESTER\_SKIP\_ERRORS

NESTER\_INCORRECT\_AREA:

NESTER\_CREATE\_INVALID\_AREA\_ERROR\_MESSAGE

INVOKE MessageBox, 0, addr NESTER\_MESSAGE, addr NESTER\_WINDOW\_TITLE, 0

NESTER\_SKIP\_ERRORS:

MOV NESTER\_MESSAGE, 0h

ENDM

NESTER\_SIXTH\_LAB:

MOV esi, 0

.WHILE esi < 8

NESTER\_CALCULATE

ADD esi, 1

.ENDW

INVOKE ExitProcess, NULL

END NESTER\_SIXTH\_LAB

**Висновки**

Під час виконання цієї лабораторної роботи я навчився обчислювати арифметичних виразів і трансцендентні функції на мові програмування Асемблер. Також використовував команди співпроцесора ix87. Познайомився з такими командами, як FDIV, FMUL, FSUB, FADD, FYL2X та інші. При написанні програми я обробив усі можливі випадки з передаванням чисел, що не входять в область допустимих значень функції логарифму та випадки з діленням на нуль, в тому числі, якщо передати a, або d, що дорівнюють нулю, програма повідомить про те, що це число недопустиме. Я порахував кожний приклад за допомогою Wolphram Mathematics, підбираючи таку ж точність, яка виходила при обчисленні у програми для кожного приклада, у результатах відрізнялася, щонайбільше, остання цифра, це точність 0.000000000000001. Отже програма рахує з достатньою точністю та не пропустить числа, які не входять в одз.