МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського

«Харківський авіаційний інститут»

Факультет радіоелектроніки, комп’ютерних систем та інфокомунікацій

Кафедра комп’ютерних систем, мереж і кібербезпеки (503)

Лабораторна робота № *3*

|  |  |
| --- | --- |
|  | Вивчення логічних команд МП Intel х86 |
|  | (назва лабораторної роботи) |
| з дисципліни | *Архітектура комп'ютерів* |
|  | (шифр)  ХАІ**.**503**.**525a**.**03О**.**123-Комп'ютерна інженерія**,** ПЗ №9629619 |

|  |  |  |
| --- | --- | --- |
| Виконав студент гр. | 525а | *Литвиненко А.В.* |
| 26.10.22 | (№ групи) | (П.І.Б.) |
| (підпис, дата) |  |  |
| Перевірив | канд. техн. наук, доцент | |
|  |  | *В. І. Дужий* |
| (підпис, дата) |  | (П.І.Б.) |

Харків – 2022

**Тема роботи:** Вивчення логічних команд МП Intel х86

**Мета роботи**:

1. Класифікація команд управління програмою.

2. Вивчення команд умовного та безумовного переходу.

3. Вивчення режимів адресації у командах розгалуження.

4. Формування машинного коду у командах розгалуження

**Варіант 5**

**Задача 1**

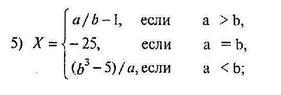
**Частина 1**. Постановка завдання

**Умова:** Написати програму, яка обчислює значення умовного висловлювання.

Дві змінні а і b, розміщені у пам'яті і є цілими знаковими числами. При обчисленні значень арифметичних виразів використовують значення, обчислені за формулами: a/(n+2), b/(n+3), де n номер варіанта.

Програма визначає відношення між числами, та обчислює значення лише одного арифметичного виразу.

**Умова з додатка:**



**Частина 2**. Схема алгоритму

**Опис на псевдокоді**

// Підрахувати початкові значення

A1 = A/(n+2);

B1 = B/(n+3);

// Помітка помилки

err=0;

X = 0;

Якщо(A1 == B1)

{

X = -25;

}

Якщо(А1 > В1){

Якщо(В1 == 0)

err = 1;

Інакше

X = a/b - 1;

}

Інакше{

Якщо(А1 == 0)

err = 1;

Інакше

X = (b\*b\*b – 5) / a;

}

Вивести X;

На основі постановки завдання розроблений алгоритм, представлений на рисунку 1.

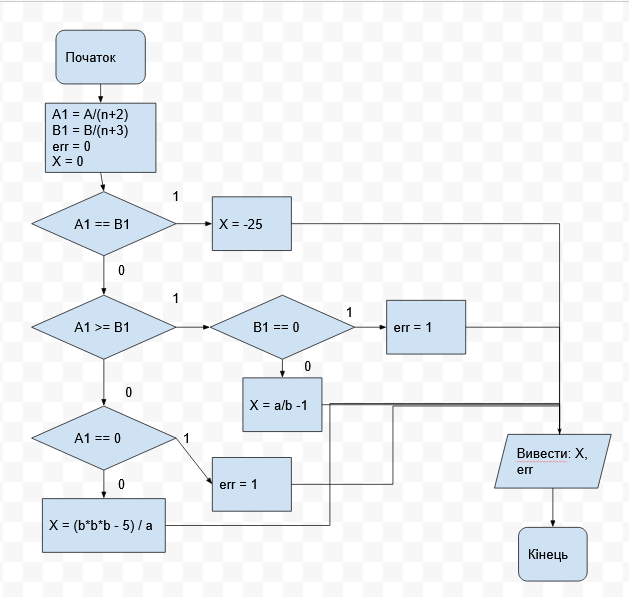


Рисунок 1 - Алгоритм перетворення

**Частина 3**. Розробка тестів

Таблиця 1 – Тестові набори

|  |  |  |
| --- | --- | --- |
| № | Вхідні дані | Очікуваний результат |
| 1 | 7 16 | 3 |
| 2 | 49 8 | 6 |
| 3 | 0 0 | -25 |
| 4 | 0 8 | Error: divided by zero |
| 5 | 7 0 | Error: divided by zero |

**Частина 4**. Текст програми

Відповідно до розробленого алгоритму в середовищі Microsoft Visual Studio була написана програма, яка наведена нижче.

#include <stdio.h>

#include <iostream>

#include <iomanip>

using namespace std;

long int

a = 0, b = 0, x = 0,

a1, b1, x1, tmp;

int err;

int err\_a;

# define ONE 1

# define FIVE 5

# define TWENTY\_FIVE 25

# define VAR 5

int main()

{

printf("\n\t\t(C) Lytvynenko Andrii, 2022");

printf("\n\t a > b: a/b-1");

printf("\n\t a = b: -25");

printf("\n\t a < b: (b\*b\*b-5)/a");

printf("\nPlease, enter A,B: ");

scanf("%li%li", &a, &b);

a1 = a / (VAR + 2);

b1 = b / (VAR + 3);

err = 0;

printf("a = %li b = %li\n", a1, b1);

if (a1 == b1) {

x = -TWENTY\_FIVE;

}

else {

if (a1 > b1) {

if (b1 == 0) {

err = 1;

}

else {

x = a1 / b1 - ONE;

}

}

else {

if (a1 == 0) {

err = 1;

}

else {

x = (b1 \* b1 \* b1 - FIVE) / a1;

}

}

}

\_\_asm {

mov err\_a, 0

}

\_\_asm {

mov ebx, VAR+3

mov eax, b

cdq

idiv ebx

mov edi, eax

mov ebx, VAR+2

mov eax, a

cdq

idiv ebx

}

\_\_asm {

cmp eax,edi

je Equal

jl Less

Greater:

// if b == 0

test edi, edi

je Error

// else

// x = a / b - ONE;

cdq

idiv edi

sub eax, ONE

mov x1, eax

jmp End

Equal:

// if a == b

mov eax, TWENTY\_FIVE

neg eax

mov x1, eax

jmp End

// if a < b

Less:

// if a == 0

test eax, eax

je Error

// x = (b1 \* b1 \* b1 - FIVE) / a;

mov tmp, edi

imul edi, edi

imul edi, tmp

sub edi, FIVE

cdq

xchg edi, eax

idiv edi

mov x1, eax

jmp End

Error:

inc err\_a

End:

// a - edi

// b - ebx

}

if (err) {

printf("[ERROR] [C] Divided by zero!\n");

}

else {

printf("[SUCCESS] [C] The result: %li\n", x);

}

if (err\_a) {

printf("[ERROR] [ASM] Divided by zero!\n");

}

else {

printf("[SUCCESS] [ASM] The result: %li!\n", x1);

}

}

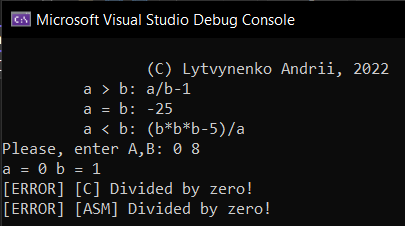
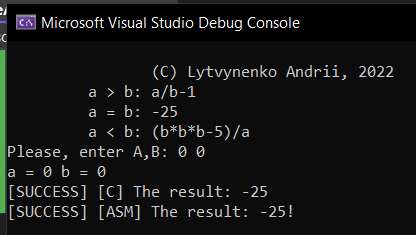
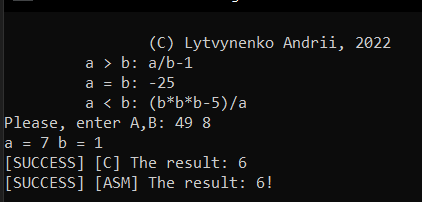
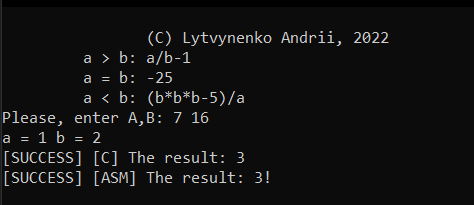
**Частина 5**. Тестування

Результати тестування наведені в таблиці 2.

Таблиця 2 – Результати тестування

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер тесту | Вхідні дані | Очікуваний результат | Отриманий результат | Результат тестування |
| 1 | 7 16 | 3 | 3 | Good |
| 2 | 49 8 | 6 | 6 | Good |
| 3 | 0 0 | -25 | -25 | Good |
| 4 | 0 8 | Error: divided by zero | Error: divided by zero | Good |
| 5 | 7 0 | Error: divided by zero | Error: divided by zero |  |

**Скриншот тестування:**



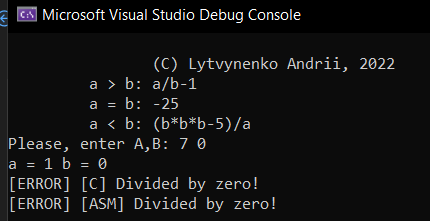


Рисунок 2 – скришот тестування

**Висновки**

Під час цієї лабораторної роботи я вдосконалив своє володіння мовою програмування асемблером та закріпив вже маючі знання, також я навчився робити розгалудження і реалізовувати їх за допомогою асемблера.