

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ (МИИТ))**

Институт транспортной техники и систем управления

Кафедра «Управление и защита информации»

отчёт По ИНДИВИДУАЛЬНОЙ работе №2

«Логические команды и команды манипулирования битами»

по дисциплине

«Машинно-ориентированные языки программирования»

**Выполнил:** студент группы ТКИ-342

Фещенко В. А.

**Проверил:** доцент кафедры УиЗИ, к.т.н. Логинова Л. Н.

**Москва 2022 г.**

**Цель работы**

Применение на практике навыков разработки программ на языке Ассемблера.

**Постановка задачи**

Реализовать вычисление заданного выражения как ассемблерную вставку в программе на языке С++.

Примечание: Все величины являются целочисленными. А,B,C –массивы. Если у массива указан один идекс, подразумевается, что он одномерный, если два индекса – двумерный. Индекс i=1,2,3.

Обработайте ситуацию возможного переполнения.

14)

**Блок схема алгоритма**



Рисунок - Блок-схема алгоритма

**Код программы**

#include <iostream>

#include <random>

#include <iomanip>

size\_t min = 0;

size\_t max = 5;

const size\_t arrSize = 3;

size\_t random()

{

std::random\_device rd;

std::mt19937 rng(rd());

std::uniform\_int\_distribution<size\_t> uni(min, max);

return uni(rng);

}

void randomArray(size\_t arr[arrSize])

{

for (size\_t i = 0; i < arrSize; i++)

arr[i] = random();

}

void randomMatrix(size\_t arr[arrSize][arrSize])

{

for (size\_t i = 0; i < arrSize; i++)

for (size\_t j = 0; j < arrSize; j++)

arr[i][j] = random();

}

size\_t main()

{

size\_t X = random();

size\_t Y = random();

size\_t term = 0;

size\_t anotherCounter = 0;

size\_t result = 0;

size\_t Bytes = 4;

bool overflow = false;

size\_t A[arrSize] = {};

size\_t C[arrSize] = {};

size\_t B[arrSize][arrSize] = {};

randomArray(A);

randomArray(C);

randomMatrix(B);

std::cout << "X: " << X << " Y: " << Y << "\n";

std::cout << "A: \n";

for (size\_t i = 0; i < arrSize; i++)

{

std::cout << std::setw(3) << A[i];

}

std::cout << "\nB:";

for (size\_t i = 0; i < arrSize; i++)

{

std::cout << "\n";

for (size\_t j = 0; j < arrSize; j++)

{

std::cout << std::setw(3) << B[i][j];

}

}

std::cout << "\nC: \n";

for (size\_t i = 0; i < arrSize; i++)

{

std::cout << std::setw(3) << C[i];

}

// result = X \* Σ(A\_i \* B\_i\_j) + Y \* ΣC\_i

\_\_asm

{

pushad

lea edx, arrSize

mov ecx, [edx]

lea esi, C

c\_sum: //Σ(C\_i)

jo except

mov eax, [esi]

add term, eax

add esi, 4

loop c\_sum

mov eax, term

mul Y //Y \* Σ(C\_i)

jo except

add result, eax

mov term, 0

//Σ(A\_i \* B\_i\_j)

lea esi, A

lea edi, B

mov ecx, arrSize

mov ebx, 0

mul\_loop:

mov eax, [esi]

mul[edi]

jo except

add term, eax

mov eax, arrSize

mul Bytes

add edi, eax

loop mul\_loop

inc anotherCounter

mov eax, anotherCounter

mul Bytes

lea edi, B

add edi, eax

add esi, 4

mov ecx, arrSize

add ecx, 1

inc ebx

cmp ebx, arrSize

jge after\_loop

loop mul\_loop

after\_loop:

mov eax, term //X \* Σ(A\_i \* B\_i\_j)

mul X

jo except

add result, eax

jmp no\_trouble

except:

add overflow, 1

no\_trouble:

popad

}

if (overflow)

{

std::cout << "\noverflow" << "\n";

}

else

{

std::cout << "\n" << result << "\n";

}

return 0;

}

**Результат выполнения программы**

*Изображение выглядит как текст

Автоматически созданное описание*

Рисунок - Результат выполнения программы

**Вывод**

В ходе выполнения Индивидуального Задания № 1 по дисциплине «Машинно-ориентированные языки программирования» были приобретены навыки разработки программ на языке ассемблера.