**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №6**

**по дисциплине «Организация ЭВМ и систем»**

Тема: **Организация связи Ассемблера с ЯВУ на примере программы построения частотного распределение попаданий псевдослучайных целых чисел в заданные интервалы.**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 1303 |  | Хабибуллина А.М. |
| Преподаватель |  | Ефремов М.А. |

Санкт-Петербург

2022

## **Цель работы.**

* 1. Научится связывать язык высокого уровня С++ с Ассемблером. Разработать программу, считывающую исходные данные, исходя из которых, генерируется массив псевдослучайных чисел и подсчитывается количество чисел в каждом из заданных интервалов.
  2. Задание.

На языке высокого уровня С++ генерируется массив псевдослучайных целых чисел, изменяющихся в заданном диапазоне и имеющих равномерное распределение.

Далее должен вызываться ассемблерный модуль(модули) для формирования распределения количества попаданий псевдослучайных целых чисел в заданные интервалы. В общем случае интервалы разбиения диапазона изменения псевдослучайных чисел могут иметь различную длину.

Результирующий массив частотного распределения чисел по интервалам, сформированный на ассемблерном уровне, возвращается в программу, реализованную на ЯВУ, и затем сохраняется в файле и выводится на экран средствами ЯВУ.

Исходные данные.

1. Длина массива псевдослучайных целыхчисел - NumRanDat (<= 16K, К=1024)

2. Диапазон изменения массива псевдослучайных целых чисел [Xmin, Xmax] , значения могут быть биполярные;

3. Количество интервалов, на которые разбивается диапазон изменения массива псевдослучайных целых чисел - NInt ( <=24 )

4. Массив левых границ интервалов разбиения LGrInt (должны принадлежать интервалу [Xmin, Xmax])

Для бригад с нечетным номером: подпрограмма формирования распределения количества попаданий псевдослучайных целых чисел в заданные интервалы реализуется в виде одного ассемблерного модуля, сразу формирующего требуемое распределение и возвращающего его в головную программу, написанную на ЯВУ.

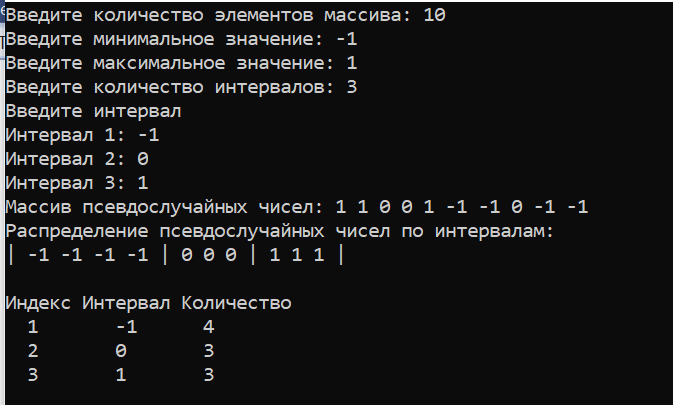
Выполнение работы

Для связи asm и C++ используется следующее объявление функции:

extern "C" {void function(int\* Array, int len, int\* LGrInt, int NInt, int\* answer); }

В программе файла lab6.cpp происходит считывание данных о количестве псевдослучайных чисел, минимальное и максимальное значение, количество интервалов и сами интервалы. После происходит генерирование псевдослучайных чисел в массив. Вызывается функция, написанная на языке Ассемблер, из файла lab6\_.asm. Сначала сохраняем значение используемых регистров в стек. В блоке loop\_ числа сортируются по интервалам, выход из блока произойдет, когда будут обработаны все числа из массива. Пока числа соответствуют интервалу, программа находится в блоке border\_count. Иначе происходит переход в блок out\_border, значение увеличится на 1 и произойдет переход к следующему значению. После выхода из блока loop\_ происходит восстановление всех используемых регистров.

**Тестирование**

****

* 1. Выводы

В ходе выполнения лабораторной работы была разработана программа генерирующая массив псевдослучайных чисел, после подсчитывающая количество чисел в каждом из заданных интервалов.

Приложение А  
Исходный код программы

Название файла: lab6.cpp

#include <iostream>

#include <fstream>

extern "C" {void function(int\* Array, int len, int\* LGrInt, int NInt, int\* answer); }

std::ofstream file("result.txt");

int main()

{

    setlocale(LC\_ALL, "Russian");

    int NumRamDat;

    int Xmin;

    int Xmax;

    int NInt;

    int\* Array;

    int\* LGrInt;

    srand(time(NULL));

    std::cout << "Введите количество элементов массива: ";

    std::cin >> NumRamDat;

    Array = new int[NumRamDat];

    std::cout << "Введите минимальное значение: ";

    std::cin >> Xmin;

    std::cout << "Введите максимальное значение: ";

    std::cin >> Xmax;

    if (Xmin >= Xmax) {

        std::cout << "Некорректные данные ;(";

        return 0;

    }

    for (int i = 0; i < NumRamDat; i++)

        Array[i] = Xmin + rand() % (Xmax - Xmin + 1);

    std::cout << "Введите количество интервалов: ";

    std::cin >> NInt;

    if (NInt < 0 || NInt > 24) {

        std::cout << "Некорректные данные ;(";

        return 0;

    }

    LGrInt = new int[NInt];

    std::cout << "Введите интервал\n";

    for (int i = 0; i < NInt; i++)

    {

        std::cout << "Интервал" << " " << i + 1 << ": ";

        std::cin >> LGrInt[i];

        if (LGrInt[i] > Xmax || LGrInt[i] < Xmin) {

            std::cout << "Некорректные данные";

            return 0;

        }

    }

    for (int i = 0; i < NInt; i++)

        for (int j = i; j < NInt; j++)

            if (LGrInt[i] > LGrInt[j])

                std::swap(LGrInt[i], LGrInt[j]);

    file << "Массив псевдослучайных чисел: ";

    std::cout << "Массив псевдослучайных чисел: ";

    for (int i = 0; i < NumRamDat; i++) {

        file << Array[i] << " ";

        std::cout << Array[i] << " ";

    }

    for (int i = 0; i < NumRamDat; i++)

        for (int j = i; j < NumRamDat; j++)

            if (Array[i] > Array[j])

                std::swap(Array[i], Array[j]);

    int\* answer = new int[NInt] {0};

    function(Array, NumRamDat, LGrInt, NInt, answer);

    std::cout << "\n";

    std::cout << "Распределение псевдослучайных чисел по интервалам:\n";

    int j = 0;

    int split = answer[j];

    if (NInt != 0) std::cout << "| ";

    for (int i = 0; i < NumRamDat; i++) {

        if (i + 1 < split || NInt == 0) std::cout << Array[i] << " ";

        else {

            j++;

            split += answer[j];

            std::cout << Array[i] << " | ";

            file << Array[i] << " | ";

        }

    }

    std::cout << "\n\n";

    std::cout << "Индекс " << "Интервал " << "Количество" << std::endl;

    file << "\n\n";

    file << "Индекс " << "Интервал " << "Количество" << std::endl;

    for (int i = 0; i < NInt; i++) {

        std::cout << "  " << i + 1 << "\t  " << LGrInt[i] << "\t  " << answer[i] << '\n';

        file << "  " << i + 1 << "\t  " << LGrInt[i] << "\t  " << answer[i] << '\n';

    }

    file.close();

    return 0;

}

Название файла: lab6\_.asm

.586p

.MODEL FLAT, C

.CODE

function PROC C USES EDI ESI, array:dword, len:dword, LGrInt:dword, NInt:dword, answer:dword

push eax

push ebx

push ecx

push edi

push esi

mov ecx, len

mov esi, array

mov edi, LGrInt

mov eax, 0

loop\_:

mov ebx, 0

border\_count:

cmp ebx, NInt

jge out\_border

push eax

mov eax, [esi + 4 \* eax]

cmp eax, [edi + 4 \* ebx]

pop eax

jl out\_border

inc ebx

jmp border\_count

out\_border:

dec ebx

cmp ebx, -1

je next

mov edi, answer

push eax

mov eax, [edi + 4 \* ebx]

inc eax

mov [edi + 4 \* ebx], eax

pop eax

mov edi, LGrInt

next:

inc eax

loop loop\_

pop esi

pop edi

pop ecx

pop ebx

pop eax

ret

function ENDP

END