

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.

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

Задание.

На языке высокого уровня C++ 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 целых чисел - NumRandat ($\leq 16K$, $K=1024$)
2. Диапазон изменения массива псевдослучайных целых чисел $[X_{min}, X_{max}]$, 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 $[X_{min}, X_{max}]$)

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 главную программу, написанную на ЯВУ.

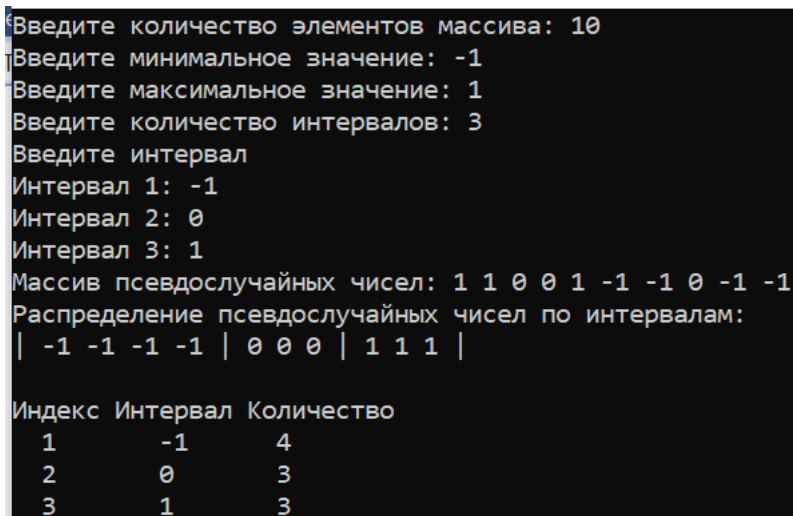
Выполнение работы

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



```
Введите количество элементов массива: 10
Введите минимальное значение: -1
Введите максимальное значение: 1
Введите количество интервалов: 3
Введите интервал
Интервал 1: -1
Интервал 2: 0
Интервал 3: 1
Массив псевдослучайных чисел: 1 1 0 0 1 -1 -1 0 -1 -1
Распределение псевдослучайных чисел по интервалам:
| -1 -1 -1 -1 | 0 0 0 | 1 1 1 |

Индекс Интервал Количество
1      -1      4
2       0      3
3       1      3
```

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
```