

**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.**

Студент гр. 0383

Самара. Р.Д.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2021

## Цель работы.

На языке C программируется ввод с клавиатуры и контроль исходных данных, а также генерируется массив псевдослучайных целых чисел, изменяющихся в заданном диапазоне и имеющих заданный закон распределения. Необходимые датчики псевдослучайных чисел находятся в каталоге RAND\_GEN (при его отсутствии получить у преподавателя).

Следует привести числа к целому виду с учетом диапазона изменения.

Далее должны вызываться 1 или 2 ассемблерных процедуры для формирования распределения количества попаданий псевдослучайных целых чисел в заданные интервалы. Ассемблерные процедуры должны вызываться как независимо скомпилированные модули. Передача параметров в процедуру должна выполняться через кадр стека.

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 ( $\leq 16K$ )
2. Диапазон изменения массива псевдослучайных целых чисел  
[Xmin, Xmax] (м.б. биполярный, например, [-100, 100])
3. Массив псевдослучайных целых чисел  $\{X_i\}$ .
4. Количество интервалов, на которые разбивается диапазон изменения массива псевдослучайных целых чисел - NInt ( $\leq 24$ )
5. Массив левых границ интервалов разбиения LGrInt .

В общем случае интервалы разбиения диапазона изменения псевдослучайных чисел могут иметь различную длину, левые границы могут задаваться в произвольном порядке и иметь произвольные значения. Если  $X_{min} < LGrInt(1)$ , то часть данных не будет участвовать в формировании распределения. Каждый интервал, кроме последнего, следует интерпретировать как  $[LGrInt(i), LGrInt(i+1))$ . Если у последнего интервала правая граница

меньше  $X_{\max}$ , то часть данных не будет участвовать в формировании распределения.

Результаты:

Текстовая таблица, строка которой содержит:

- номер интервала,
- левую границу интервала,
- количество псевдослучайных чисел, попавших в интервал.

Количество строк должно быть равно числу интервалов разбиения.

Таблица должна выводиться на экран и сохраняться в файле.

### **Задание.**

Вариант 13. Вид распределения случайных чисел равномерный, одна ассемблерная процедура.  $N_{\text{int}} \geq D_x$ ,  $Lg1 \leq X_{\min}$ ,  $ПГ_{\text{посл}} > X_{\max}$

### **Выполнение работы.**

Была создана программа на языке C++, использующая ассемблерную процедуру. Реализован ввод с клавиатуры и контроль исходных данных, а также генерация массив псевдослучайных целых чисел, изменяющихся в заданном диапазоне и имеющих заданный закон распределения.

В ассемблерном модуле реализована процедура FUNC, которая принимает указатель на массив псевдосгенерированных чисел и его размер, указатель на массив левых границ интервалов и его размер, а также указатель на результирующий массив. Внутри процедуры циклически с начала массива для каждого числа проверяется принадлежность к интервалу (ищутся с конца), и если это так, в результирующем массиве увеличивается количество чисел, принадлежащих данному интервалу на 1.

В конце происходит вывод результирующей таблицы на экран и ее запись в файл (out.txt).

Тексты исходных файлов программ см. в приложении А.

## Тестирование.

Результаты тестирования представлены в виде серии изображений.

### Изображение 1 – Результаты тестирования

```
Введите количество генерируемых псевдослучайных целых чисел (от 0 по 16000): -5
Количество чисел должно быть от 0 по 16000
D:\OrgEvm\lb6\lab6\Debug\lab6.exe (процесс 14520) завершил работу с кодом -1.
Нажмите любую клавишу, чтобы закрыть это окно...
```

### Изображение 2 – Результаты тестирования

```
Введите количество генерируемых псевдослучайных целых чисел (от 0 по 16000): 20
Введите минимальное значение (левую границу): -5
Введите максимальное значение (правую границу): 5
Количество интервалов должно быть не меньше диапазона изменения входных чисел
Введите количество интервалов, на которые разбивается диапазон изменения массива псевдослучайных целых чисел (от 1 по 24): 10
Предупреждение:
Самая маленькая левая граница интервалов разбиения должна быть меньше либо равна минимальному значению
Введите левые границы интервалов разбиения: -7 2 3 4 1 0 -2 -3 4 -4
Сгенерированные числа: 3 -5 3 -3 0 -4 -2 -1 -1 2 1 -5 3 0 5 2 -4 -3 2 2
Номер интервала.    Левая граница интервала.    Количество чисел в интервале.
1                   -7                           2
2                   -4                           2
3                   -3                           2
4                   -2                           3
5                   0                            2
6                   1                            1
7                   2                            4
8                   3                            3
9                   4                            0
10                  4                            1
Для продолжения нажмите любую клавишу . . .
```

### Изображение 3 – Результаты тестирования

```
Введите количество генерируемых псевдослучайных целых чисел (от 0 по 16000): 10
Введите минимальное значение (левую границу): 0
Введите максимальное значение (правую границу): 10
Количество интервалов должно быть не меньше диапазона изменения входных чисел
Введите количество интервалов, на которые разбивается диапазон изменения массива псевдослучайных целых чисел (от 1 по 24): 15
Предупреждение:
Самая маленькая левая граница интервалов разбиения должна быть меньше либо равна минимальному значению
Введите левые границы интервалов разбиения: -1 0 1 6 9 2 3 6 7 2 4 6 7 9 0
Сгенерированные числа: 7 1 10 0 4 5 1 6 9 6
Номер интервала.    Левая граница интервала.    Количество чисел в интервале.
1                   -1                           0
2                   0                            0
3                   0                            1
4                   1                            2
5                   2                            0
6                   2                            0
7                   3                            0
8                   4                            2
9                   6                            0
10                  6                            0
11                  6                            2
12                  7                            0
13                  7                            1
14                  9                            0
15                  9                            2
Для продолжения нажмите любую клавишу . . .
```

## **Вывод.**

Была изучена организация связи Ассемблера с ЯВУ на примере программы построения частотного распределение попаданий псевдослучайных целых чисел в заданные интервалы на языке C++ с использованием ассемблерного модуля.

## **ПРИЛОЖЕНИЕ А**

### **ТЕКСТЫ ИСХОДНЫХ ФАЙЛОВ ПРОГРАММ**

Название файла: Lab6\_asm.asm

```
.586
.MODEL FLAT, C
.CODE

FUNC PROC C NumRanDat:dword, Xi:dword, NInt:dword, LGrInt:dword, result:dword
    push eax
    push ebx
    push ecx
    push esi
    push edi

    mov ecx, NumRanDat
    mov esi, Xi
    mov edi, LGrInt      ; левые границы
    mov eax, 0           ; eax - индекс текущего числа

loop1:
    mov ebx, 0           ; ebx - индекс текущего интервала
brders:
    cmp ebx, NInt
    jge boarders_end
    push eax
    mov eax, [esi+4*eax]
    cmp eax, [edi+4*ebx]
    pop eax
    jl boarders_end
    inc ebx
    jmp brders

boarders_end:
    dec ebx              ; -1
    cmp ebx, -1
    je skip
    mov edi, result
    push eax
    mov eax, [edi+4*ebx]
    inc eax
    mov [edi+4*ebx], eax
    pop eax
    mov edi, LGrInt
```

```

        skip:
            inc eax        ; eax++
        loop loop1    ;cx != 0

    pop ebx
    pop eax
    pop edi
    pop esi
    pop ecx
    ret
FUNC ENDP
END

```

### Название файла: lab6.cpp

```

#include <iostream>
#include <fstream>
#include <random>
#include <ctime>
#include <algorithm>

extern "C" void FUNC(int NumRanDat, int* Xi, int NInt, int* LGrInt, int*
result);

int main() {
    setlocale(LC_ALL, "");
    int NumRanDat;
    std::cout << "Введите количество генерируемых псевдослучайных целых чисел (от 0 по
16000): ";
    std::cin >> NumRanDat;
    if (NumRanDat <= 0 || NumRanDat > 16000) {
        std::cout << "Количество чисел должно быть от 0 по 16000";
        return -1;
    };
    int Xmin, Xmax;
    std::cout << "Введите минимальное значение (левую границу): ";
    std::cin >> Xmin;
    std::cout << "Введите максимальное значение (правую границу): ";
    std::cin >> Xmax;
    if (Xmax <= Xmin) {
        std::cout << "Неверно введены максимальное и минимальное значения";
        return -1;
    }
    int NInt;
    std::cout << "Количество интервалов должно быть не меньше диапазона изменения входных
чисел" << std::endl;
    std::cout << "Введите количество интервалов, на которые разбивается диапазон изменения
массива псевдослучайных целых чисел (от 1 по 24): ";
    std::cin >> NInt;
    if (NInt <= 0 || NInt > 24 || NInt < Xmax - Xmin) {
        std::cout << "Неверно введено количество интервалов";
        return -1;
    }
    int* LGrInt = new int[NInt];
    std::cout << "Предупреждение:" << std::endl;
    std::cout << "Самая маленькая левая граница интервалов разбиения должна быть меньше
либо равна минимальному значению" << std::endl;
    std::cout << "Введите левые границы интервалов разбиения: ";

```

```

for (int i = 0; i < NInt; ++i)
    std::cin >> LGrInt[i];

for (int i = 0; i < NInt - 1; i++) {
    for (int j = i + 1; j < NInt; j++) {
        if (LGrInt[j] < LGrInt[i]) {
            std::swap(LGrInt[j], LGrInt[i]);
        }
    }
}

if (LGrInt[0] > Xmin) {
    std::cout << "Неверно введены левые границы интервалов разбиения";
    return -1;
}

std::mt19937 gen(time(nullptr));
std::uniform_int_distribution<int> dis(Xmin, Xmax);
int* Xi = new int[NumRanDat];
for (int i = 0; i < NumRanDat; ++i) Xi[i] = dis(gen);

int* result = new int[NInt];
for (int i = 0; i < NInt; ++i) result[i] = 0;

FUNC(NumRanDat, Xi, NInt, LGrInt, result);

std::ofstream file("out.txt");
file << "Сгенерированные числа: ";
std::cout << "Сгенерированные числа: ";
for (int i = 0; i < NumRanDat; ++i) {
    file << Xi[i] << ' ';
    std::cout << Xi[i] << ' ';
}
std::cout << "\nНомер интервала.\tЛевая граница интервала.\tКоличество чисел в
интервале." << std::endl;
file << "\nНомер интервала.\tЛевая граница интервала.\tКоличество чисел в интервале."
<< std::endl;
for (int i = 0; i < NInt; ++i) {
    std::cout << "\t" << i + 1 << "\t\t\t" << LGrInt[i] << "\t\t\t" <<
result[i] << std::endl;
    file << "\t" << i + 1 << "\t\t\t" << LGrInt[i] << "\t\t\t" << result[i]
<< std::endl;
}
file.close();
delete[] LGrInt;
delete[] Xi;
delete[] result;
system("pause");
return 0;
}

```