Учреждение образования

«Белорусский государственный технологический университет»

**Лабораторная работа №1**

Вспомогательные функции

Выполнил:

Студент 2 курса 1 группы ФИТ

Шумова Елизавета Игоревна

**2022 г.**

**Цель работы:** приобретение навыков составления и отладки программ с использованием пользовательских функций для замера продолжительности процесса вычисления.

Auxil.h:

#pragma once

#include "pch.h"

#include <cstdlib>

#include <ctime>

namespace auxil

{

void start(); //установка нач.числа для генератора псевдослучайных чисел

double dget(double rmin, double rmax); //генерация действительного псевдослуч.числа в заданном

int iget(int rmin, int rmax); //генерация целого псевдослучайного числа в заданном диапазоне

}

Auxil.cpp:

#include "pch.h"

#include "Auxil.h"

//RAND\_MAX = 32767

namespace auxil

{

void start() //старт генератора

{

srand((unsigned)time(NULL));

}

double dget(double rmin, double rmax)

{

return (double)rand() / RAND\_MAX \* (rmax - rmin) + rmin;

}

int iget(int rmin, int rmax)

{

//return -(rand() / RAND\_MAX \* (rmax - rmin) + rmin);

return (int)dget((double)rmin, (double)rmax);

}

}

pch.h:

#ifndef PCH\_H

#define PCH\_H

// TODO: add headers that you want to pre-compile here

#endif //PCH\_H

LR1.cpp:

#include <tchar.h>

#include "Auxil.h" // вспомогательные функции

#include <iostream>

#include <ctime>

#include <locale>

#define CYCLE 1000000 // количество циклов

int \_tmain(int argc, \_TCHAR\* argv[])

{

double av1 = 0, av2 = 0;

clock\_t t1 = 0, t2 = 0;

setlocale(LC\_ALL, "rus");

auxil::start(); // старт генерации

t1 = clock(); // фиксация времени

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

{

av1 += (double)auxil::iget(-100, 100); // сумма случайных чисел

av2 += auxil::dget(-100, 100); // сумма случайных чисел

}

t2 = clock(); // фиксация времени

std::cout << std::endl << "количество циклов: " << CYCLE;

std::cout << std::endl << "среднее значение (int): " << av1 / CYCLE;

std::cout << std::endl << "среднее значение (double): " << av2 / CYCLE;

std::cout << std::endl << "продолжительность (у.е): " << (t2 - t1);

std::cout << std::endl << " (сек): "

<< ((double)(t2 - t1)) / ((double)CLOCKS\_PER\_SEC);

std::cout << std::endl;

system("pause");

return 0;

}

Результат выполнения:

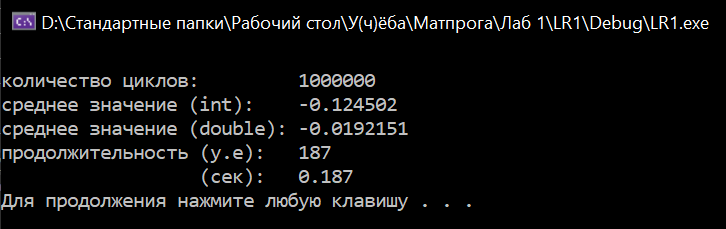


График Excel:

|  |  |  |
| --- | --- | --- |
| Количество циклов | Процессорное время | Y |
| 100000 | 21 | 27,4 |
| 200000 | 58 | 47,4 |
| 300000 | 61 | 67,4 |
| 400000 | 78 | 87,4 |
| 500000 | 104 | 107,4 |
| 600000 | 115 | 127,4 |
| 700000 | 139 | 147,4 |
| 800000 | 150 | 167,4 |
| 900000 | 169 | 187,4 |
| 1000000 | 206 | 207,4 |
|  |  |  |
| Дисперсия | 3249,877778 | 3666,666667 |
| F набл. | 1,128247558 |  |
| alpha | 0,05 |  |
| F табличное | 3,18 |  |
| F крит | 3,178893104 |  |