## **Лабораторная работа 1. Вспомогательные функции**

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

#include <iostream>

#include <ctime>

#include <random>

#include <locale>

#define CYCLE 1000000

void start() {

srand(time(NULL));

}

int dget(int rmin, int rmax) {

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

}

int iget(int rmin, int rmax) {

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

}

long double fact(int N)

{

if (N < 0) return 0;

if (N == 0) return 1;

else return N \* fact(N - 1);

}

int f(int n)

{

if (n == 1 || n == 2) return (n - 1);

return f(n - 1) + f(n - 2);

}

int main() {

double av1 = 0, av2 = 0;

clock\_t t1 = 0, t2 = 0;

setlocale(LC\_ALL, "ru");

start();

t1 = clock();

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

{

av1 += (double)iget(-100, 100);

av2 += dget(-100, 100);

}

t2 = clock();

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;}

Рисунок 1 - Код программы

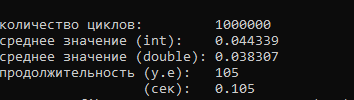


Рисунок 2 - Выполнение программы

#include <iostream>

#include <ctime>

#include <random>

#include <locale>

int f(int n) {

if (n == 1 || n == 2) return (n - 1);

return f(n - 1) + f(n - 2);

}

int main() {

setlocale(LC\_ALL, "ru");

for (int i = 30; i < 45; i++) {

std::cout << "число №" << i;

t1 = clock();

long double num = f(i);

t2 = clock();

std::cout << " fib " << num;

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

}

return 0;

}

Рисунок 4 – Код Чисел Фибоначчи

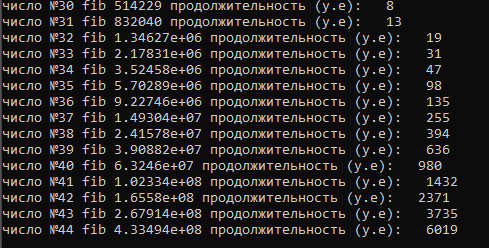


Рисунок 5 - Выполнение программы чисел Фибоначчи

**Графики продолжительности вычислений в условных единицах процессорного времени**

Рисунок 3 - График зависимости продолжительности процесса вычисления генерации случайных чисел от количества циклов

Рисунок 4 – График зависимости продолжительности процесса вычисления числа Фибоначчи от номера числа в последовательности

Вывод по графикам: В 1 эксперименте линейная зависимость, где увеличение количества циклов влияет на продолжительность процесса. Во 2 – экспоненциальная зависимость представлена кривой на графике. В случае с Фибоначчи начиная с 30, время работы увеличивается.