Задание 1.

Программа записывает в файл большое количество значений и считает время выполнения.

Программа:

#include <iostream>

#include <windows.h>

#include <fstream>

int main()

{

setlocale(LC\_ALL, "Russian");

int StartMs = GetTickCount();

std::ofstream ofs("1.txt");

for (int i = 0; i < 600000; ++i)

ofs << i << ' ';

int EndMs = GetTickCount();

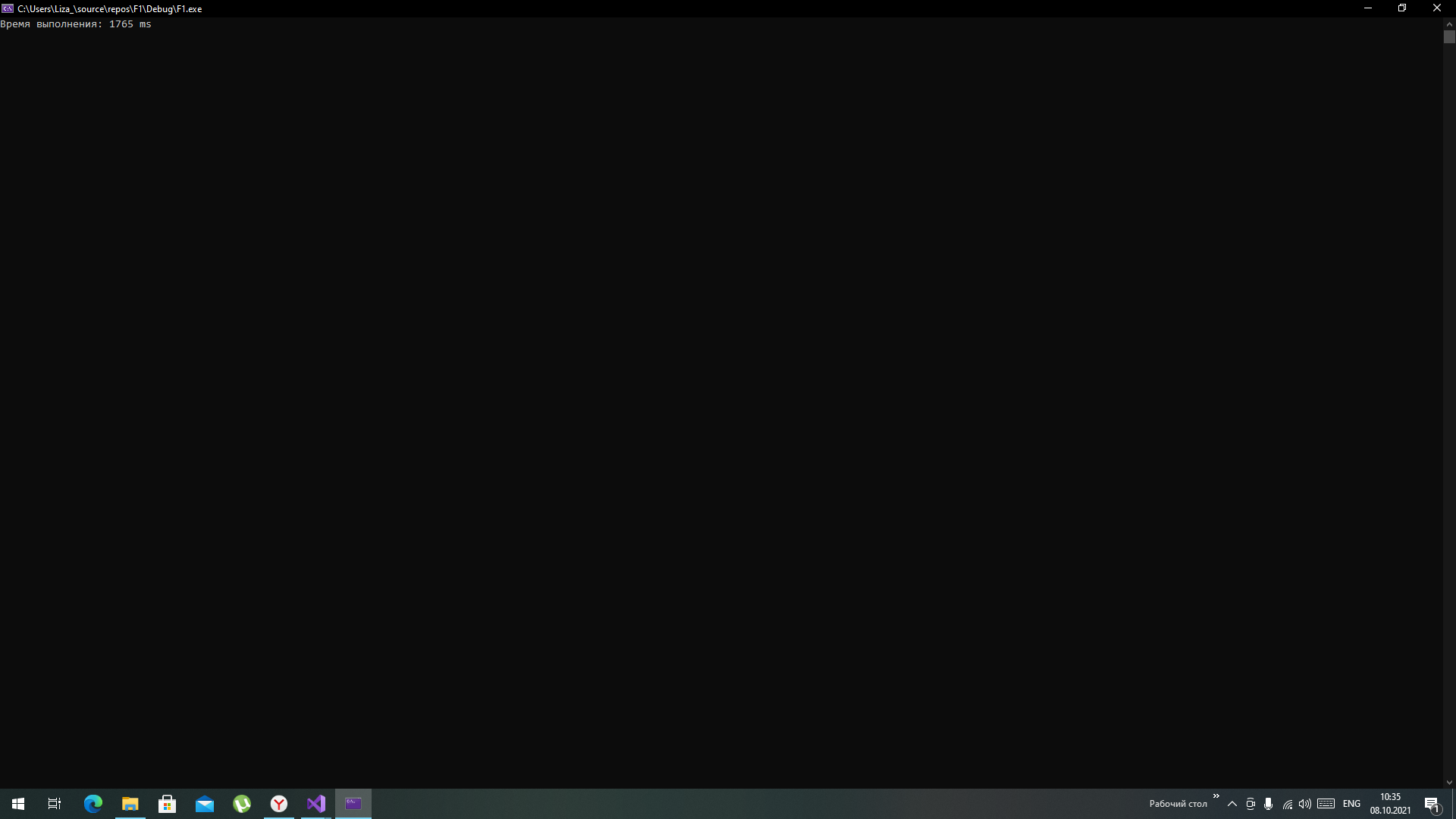
std::cout << "Время выполнения: " << EndMs - StartMs << " ms\n";

std::getchar();

return 0;

}

Время выполнения:



Задание 2.

Программа записывает в файл тот же объём информации используя потоки.

1 запуск – 6 потоков.

Программа:

#include <iostream>

#include <thread>

#include <chrono>

#include <windows.h>

#include <fstream>

using namespace std;

void DoWork()

{

std::ofstream ofs("2.txt");

for (int i = 0; i < 100000; ++i)

ofs << i << ' ';

}

int main()

{

setlocale(LC\_ALL, "ru");

int StartMs = GetTickCount();

thread th2(DoWork);

thread th3(DoWork);

thread th4(DoWork);

thread th5(DoWork);

thread th6(DoWork);

std::ofstream ofs("2.txt");

for (int i = 0; i < 100000; ++i)

ofs << i << ' ';

th2.join();

th3.join();

th4.join();

th5.join();

th6.join();

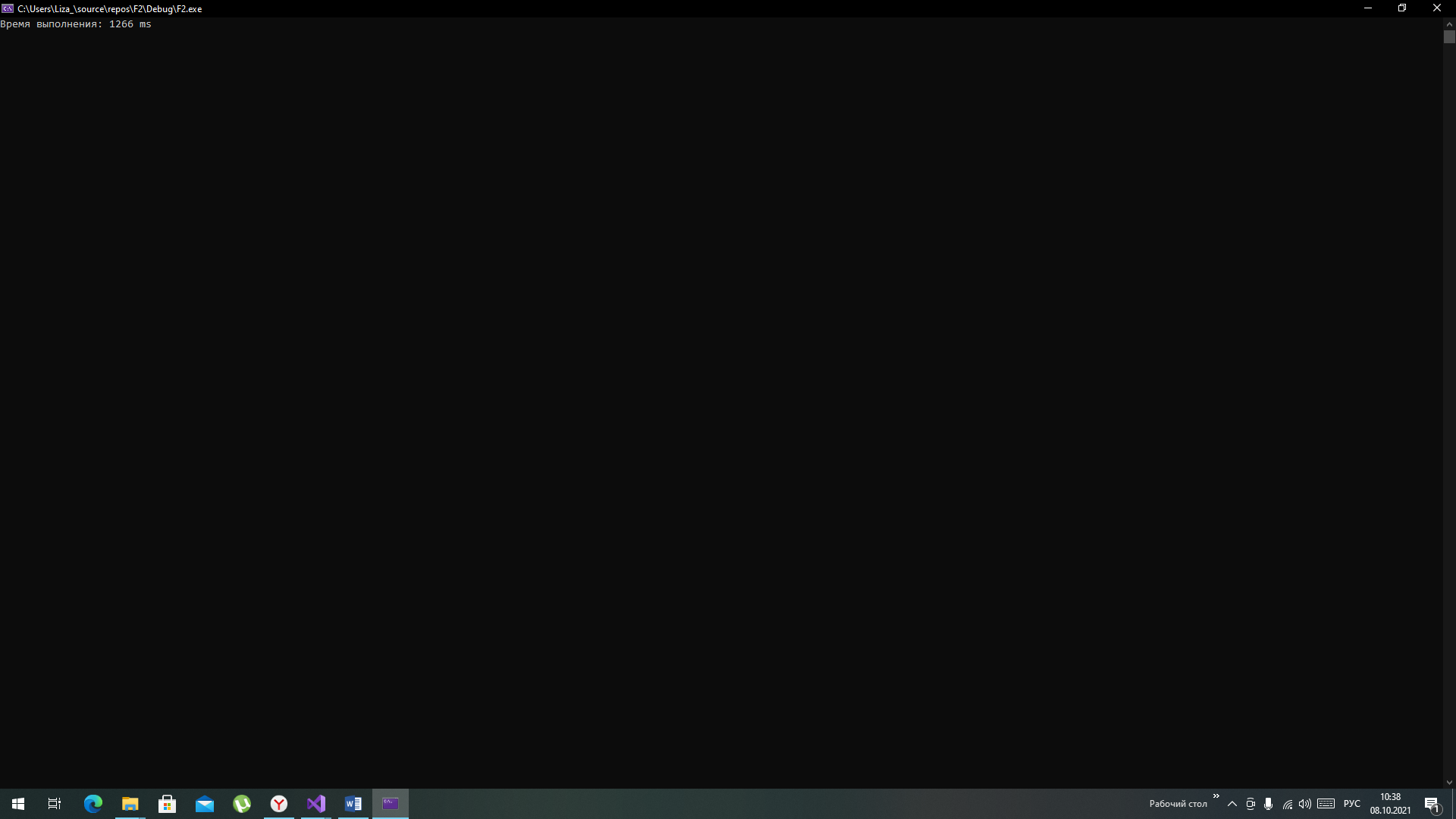
int EndMs = GetTickCount();

std::cout << "Время выполнения: " << EndMs - StartMs << " ms\n";

std::getchar();

return 0;

Время выполнения:



2 запуск – 12 потоков.

Программа:

#include <iostream>

#include <thread>

#include <chrono>

#include <windows.h>

#include <fstream>

using namespace std;

void DoWork()

{

std::ofstream ofs("2.txt");

for (int i = 0; i < 100000; ++i)

ofs << i << ' ';

}

int main()

{

setlocale(LC\_ALL, "ru");

int StartMs = GetTickCount();

thread th2(DoWork);

thread th3(DoWork);

thread th4(DoWork);

thread th5(DoWork);

thread th6(DoWork);

thread th7(DoWork);

thread th8(DoWork);

thread th9(DoWork);

thread th10(DoWork);

thread th11(DoWork);

thread th12(DoWork);

std::ofstream ofs("2.txt");

for (int i = 0; i < 100000; ++i)

ofs << i << ' ';

th2.join();

th3.join();

th4.join();

th5.join();

th6.join();

th7.join();

th8.join();

th9.join();

th10.join();

th11.join();

th12.join();

int EndMs = GetTickCount();

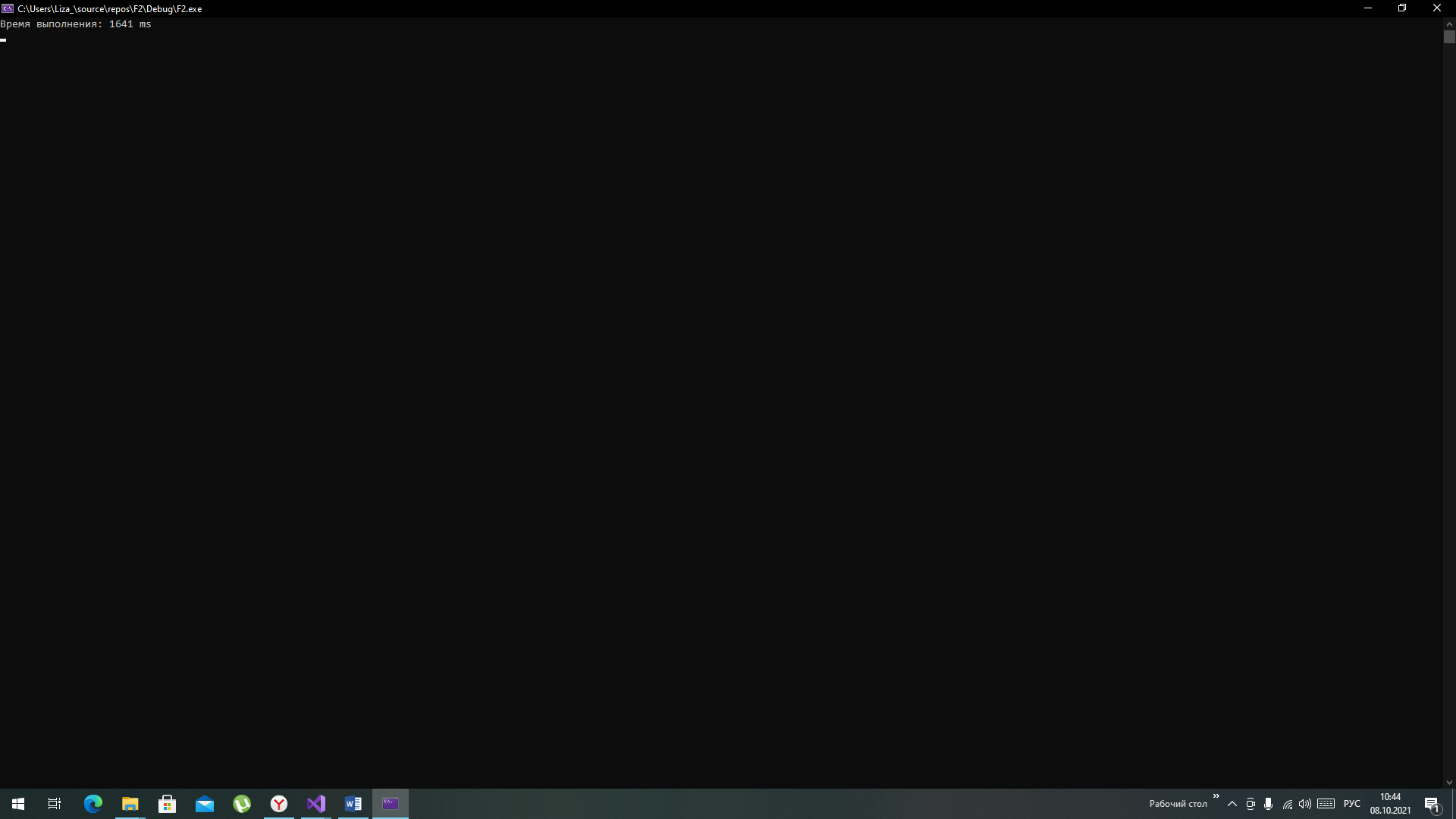
std::cout << "Время выполнения: " << EndMs - StartMs << " ms\n";

std::getchar();

return 0;

}

Время запуска:



Задание 3.

Вывод.

Программа с 6-ю потоками быстрее программы с 12-б потоками. Самая долгая по времени программа – программа с одним потоком, т.к. одному потоку приходится обрабатывать весь объём информации.