

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**ОТЧЕТ ПО ПРАКТИКЕ WORK10**

**ОТЧЕТ**

Студента 3 курса 311 группы  
направления 02.03.02 — Фундаментальная информатика и информационные  
технологии  
факультета КНИИТ  
Забоева Максима Владиславовича

Проверил

Старший преподаватель

\_\_\_\_\_

М. С. Портенко

Саратов 2023

## СОДЕРЖАНИЕ

1	Условие задачи .....	3
2	Практическая часть .....	4
2.1	Последовательная реализация .....	4
2.2	Параллельная реализация вариант 1 .....	5
2.3	Параллельная реализация вариант 2 .....	5
3	Результаты работы .....	7
3.1	Характеристики компьютера .....	7
3.2	Фото результатов .....	7

## 1 Условие задачи

Аналогично работе с OMP выполните следующее задание через MPI.

$$\int_0^{16} \int_0^{16} \frac{e^{\sin \pi x \cos \pi y} + 1}{(b_1 - a_1)(b_2 - a_2)} dx dy \approx 2.130997$$

Модифицируйте разработанную ранее программу по методу прямоугольников для численного интегрирования тестовой функции:

Сравните время численного интегрирования для последовательной и параллельных реализаций и параллельных реализаций между собой.

## 2 Практическая часть

### 2.1 Последовательная реализация

Код программы:

```
#include <iostream>
#include <mpi.h>
#include <time.h>
#include <cmath>
#define PI 3.1415926535897932384626433832795
using namespace std;
void integral(const double a, const double b, const double h, double* res)
{
    double a1 = a;
    double a2 = a;
    double b1 = b;
    double b2 = b;
    int i, n;
    double sum; // локальная переменная для подсчета интеграла
    double x; // координата точки сетки
    double y;
    n = (int)((b - a) / h); // количество точек сетки интегрирования
    int m = (int)((b - a) / h);
    sum = 0.0;
    int commsize;
    int rank;
    double Result;
    //MPI_Init(NULL, NULL);
    //MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    //MPI_Comm_size(MPI_COMM_WORLD, &commsize);
    //MPI_Bcast(&m, 1, MPI_INT, 0, MPI_COMM_WORLD);
    for (i = 0; i < n; i++)
    {
        double xi = a1 + (i * h) + (h / 2.0);

        for (int j = 0; j < m; j++)
        {
            double yi = a2 + (j * h) + (h / 2.0);
            sum = sum + ((exp(sin(PI * xi) * cos(PI * yi)) + 1) / ((b1 - a1) * (b2 - a2))) * (h * h);
        }
    }
    /* MPI_Reduce(&sum, &Result, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
    MPI_Finalize();*/
    *res = sum;
}

int main()
{
    double min_time; // минимальное время работы
    // реализации алгоритма
    double max_time; // максимальное время работы
    // реализации алгоритма
    double avg_time; // среднее время работы
    double res;
    double a = 0.0;
    double b = 16.0;
    double h = 0.1;
    int n = 6;
    min_time = clock();
    integral(a, b, h, &res);
```

```

    max_time = clock();
    avg_time = (max_time - min_time) / CLOCKS_PER_SEC;
    cout << "execution time : " << avg_time << endl;
    cout.precision(8);
    cout << "integral value : " << res << endl;
    return 0;
}

```

## 2.2 Параллельная реализация вариант 1

Код программы:

```

void integral(const double a, const double b, const double h, double* res)
{
    double a1 = a;
    double a2 = a;
    double b1 = b;
    double b2 = b;
    int i, n;
    double sum; // локальная переменная для подсчета интеграла
    double x; // координата точки сетки
    double y;
    n = (int)((b - a) / h); // количество точек сетки интегрирования
    int m = (int)((b - a) / h);
    sum = 0.0;
    int commsize;
    int rank;
    double Result;
    MPI_Init(NULL, NULL);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &commsize);
    MPI_Bcast(&n, 1, MPI_INT, 0, MPI_COMM_WORLD);
    for (i = 0; i < n; i++)
    {
        double xi = a1 + (i * h) + (h / 2.0);

        for (int j = 0; j < m; j++)
        {
            double yi = a2 + (j * h) + (h / 2.0);
            sum = sum + ((exp(sin(PI * xi) * cos(PI * yi)) + 1) / ((b1 - a1) * (b2 - a2))) * (h * h);
        }
    }
    MPI_Reduce(&sum, &Result, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
    MPI_Finalize();
    *res = sum;
}

```

## 2.3 Параллельная реализация вариант 2

Код программы:

```

void integral(const double a, const double b, const double h, double* res)
{
    double a1 = a;
    double a2 = a;
    double b1 = b;
    double b2 = b;
    int i, n;

```

```

double sum; // локальная переменная для подсчета интеграла
double x; // координата точки сетки
double y;
n = (int)((b - a) / h); // количество точек сетки интегрирования
int m = (int)((b - a) / h);
sum = 0.0;
int commsize;
int rank;
double Result;
MPI_Init(NULL, NULL);
MPI_Comm_rank(MPI_COMM_WORLD, &rank);
MPI_Comm_size(MPI_COMM_WORLD, &commsize);
MPI_Bcast(&m, 1, MPI_INT, 0, MPI_COMM_WORLD);
for (i = 0; i < n; i++)
{
    double xi = a1 + (i * h) + (h / 2.0);

    for (int j = 0; j < m; j++)
    {
        double yi = a2 + (j * h) + (h / 2.0);
        sum = sum + ((exp(sin(PI * xi) * cos(PI * yi)) + 1) / ((b1 - a1) * (b2 - a2))) * (h * h);
    }
}
MPI_Reduce(&sum, &Result, 1, MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
MPI_Finalize();
*res = sum;
}

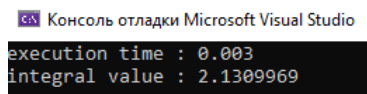
```

### 3 Результаты работы

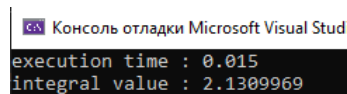
#### 3.1 Характеристики компьютера

Процессор — 12th Gen Intel Core i5-12600KF, Базовая скорость 3,70ГГц,  
Кол-во ядер 10, Кол-во процессоров 16 (включая 4 энергоэффективных ядра).  
16гб Оперативной памяти, скорость 3200МГц

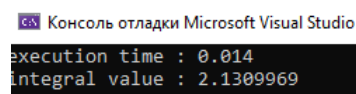
#### 3.2 Фото результатов



```
Консоль отладки Microsoft Visual Studio  
execution time : 0.003  
integral value : 2.1309969
```



```
Консоль отладки Microsoft Visual Stud  
execution time : 0.015  
integral value : 2.1309969
```



```
Консоль отладки Microsoft Visual Studio  
execution time : 0.014  
integral value : 2.1309969
```