МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №1
по дисциплине «Параллельные алгоритмы»
Тема: Запуск параллельной программы на различном числе одновременно работающих процессов, упорядочение вывода результатов.

Студент гр. 3388 _____ Дубровин Д.Н.

Преподаватель _____ Татаринов Ю.С.

Санкт-Петербург
2025

Цель работы.

Написать параллельную программу с использованием MPI, исследовать зависимость скорости выполнения программы от числа процессов, построить графики и сети Петри.

Задание.

Задание 1: 1. Все процессы, кроме процесса с рангом 0, передают значние своего ранга нулевому процессу. Процесс с рангом 0 сначала печатает
значение своего ранга, а далее принимает сообщения с рангами процессов и также печатает их значения. При этом важно отметить, что порядок приема сообщений заранее не определен и зависит от условий выполнения параллельной программы (более того, этот порядок может изменяться от запуска к запуску).

2. Проанализировать порядок вывода сообщений на экран. Для этого запустит программу несколько раз при фиксированном числе процессов, например -8ми. Вывести правило, определяющее порядок вывода сообщений.

3. Выполнить требования к содержанию отчета для задания 1

Задание 2: Модифицировать программу таким образом, чтобы порядок вывода сообщений на экран соответствовал номеру соответствующего процесса.

1. Выполнить требования к содержанию отчета для задания 2

2. Сравнить результаты работы двух программ.

Выполнение работы.

1. Процесс с номером 0 запускает таймер, выводит свой номер и начинает ожидать сообщения от всех остальных процессов. Каждый из других процессов отправляет нулевому свой ранг с помощью MPI_Send. Когда процесс 0 получает сообщение через MPI_Recv, он выводит информацию о том, от какого процесса и какой номер был получен. После получения всех сообщений процесс 0 останавливает таймер и выводит общее время выполнения программы, показывая количество задействованных процессов и затраченное время. Все процессы завершают работу после вызова MPI_Finalize.
2. Построим схему Петри для первого задания:

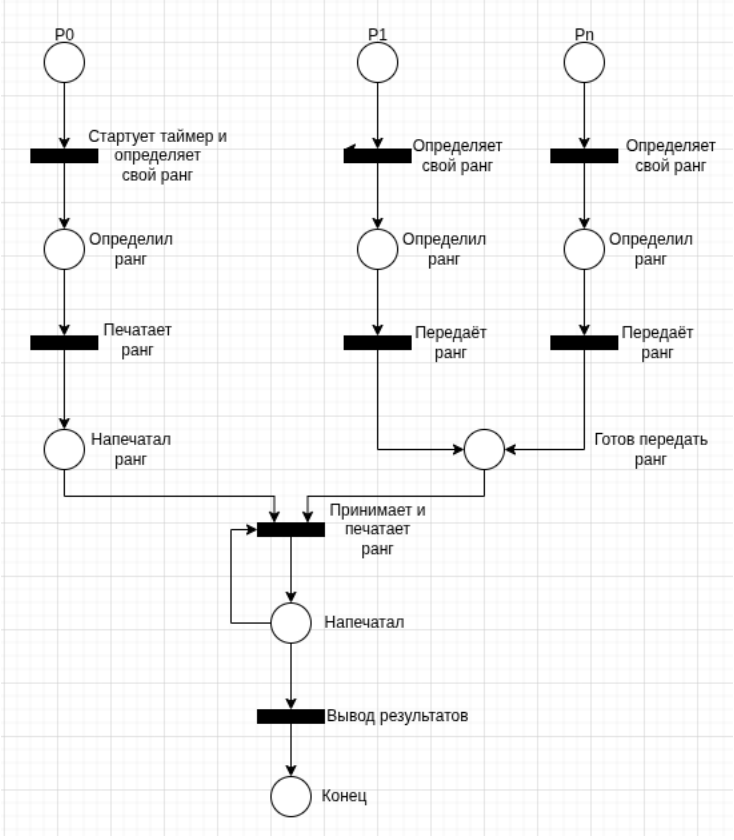

Рис.1 - сеть Петри для первой программы

Таблица 1- результаты выполнения программы.

N	2	4	6	8	10	12	14	16
t, сек	0.000078	0.000072	0.000116	0.000070	0.000067	0.000081	0.000175	0.000106

Результаты выполнения программы:

```
Запуск с 2 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
============= Execution Results =============
Num of proc: 2 | Execution time: 0.000078 seconds
---
Запуск с 3 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
============= Execution Results =============
Num of proc: 3 | Execution time: 0.000037 seconds
---
Запуск с 4 процессами...
Process 0: my rank is 0
Process 0: received rank 3 from process 3
Process 0: received rank 2 from process 2
Process 0: received rank 1 from process 1
============= Execution Results =============
Num of proc: 4 | Execution time: 0.000072 seconds
---
Запуск с 5 процессами...
Process 0: my rank is 0
Process 0: received rank 2 from process 2
Process 0: received rank 1 from process 1
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
============= Execution Results =============
Num of proc: 5 | Execution time: 0.000178 seconds
---
Запуск с 6 процессами...
Process 0: my rank is 0
Process 0: received rank 2 from process 2
Process 0: received rank 5 from process 5
Process 0: received rank 1 from process 1
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
============= Execution Results =============
Num of proc: 6 | Execution time: 0.000116 seconds
---
Запуск с 7 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 3 from process 3
============= Execution Results =============
Num of proc: 7 | Execution time: 0.000065 seconds
---
Запуск с 8 процессами...
Process 0: my rank is 0
Process 0: received rank 4 from process 4
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
```

```
Process 0: received rank 3 from process 3
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
============= Execution Results =============
Num of proc: 8 | Execution time: 0.000070 seconds
---
Запуск с 9 процессами...
Process 0: my rank is 0
Process 0: received rank 3 from process 3
Process 0: received rank 1 from process 1
Process 0: received rank 4 from process 4
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 2 from process 2
Process 0: received rank 5 from process 5
============= Execution Results =============
Num of proc: 9 | Execution time: 0.000081 seconds
---
Запуск с 10 процессами...
Process 0: my rank is 0
Process 0: received rank 9 from process 9
Process 0: received rank 1 from process 1
Process 0: received rank 3 from process 3
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 4 from process 4
Process 0: received rank 2 from process 2
============= Execution Results =============
Num of proc: 10 | Execution time: 0.000067 seconds
---
Запуск с 11 процессами...
Process 0: my rank is 0
Process 0: received rank 4 from process 4
Process 0: received rank 10 from process 10
Process 0: received rank 1 from process 1
Process 0: received rank 8 from process 8
Process 0: received rank 6 from process 6
Process 0: received rank 2 from process 2
Process 0: received rank 7 from process 7
Process 0: received rank 9 from process 9
Process 0: received rank 5 from process 5
Process 0: received rank 3 from process 3
============= Execution Results =============
Num of proc: 11 | Execution time: 0.000101 seconds
---
Запуск с 12 процессами...
Process 0: my rank is 0
Process 0: received rank 10 from process 10
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
```

```
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 11 from process 11
============= Execution Results =============
Num of proc: 12 | Execution time: 0.000081 seconds
---
Запуск с 13 процессами...
Process 0: my rank is 0
Process 0: received rank 8 from process 8
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 12 from process 12
============= Execution Results =============
Num of proc: 13 | Execution time: 0.000105 seconds
---
Запуск с 14 процессами...
Process 0: my rank is 0
Process 0: received rank 11 from process 11
Process 0: received rank 2 from process 2
Process 0: received rank 4 from process 4
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 12 from process 12
Process 0: received rank 13 from process 13
Process 0: received rank 3 from process 3
Process 0: received rank 1 from process 1
Process 0: received rank 5 from process 5
Process 0: received rank 8 from process 8
============= Execution Results =============
Num of proc: 14 | Execution time: 0.000175 seconds
---
Запуск с 15 процессами...
Process 0: my rank is 0
Process 0: received rank 12 from process 12
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 13 from process 13
Process 0: received rank 14 from process 14
============= Execution Results =============
Num of proc: 15 | Execution time: 0.000079 seconds
```

```
---
Запуск с 16 процессами...
Process 0: my rank is 0
Process 0: received rank 11 from process 11
Process 0: received rank 1 from process 1
Process 0: received rank 3 from process 3
Process 0: received rank 5 from process 5
Process 0: received rank 10 from process 10
Process 0: received rank 13 from process 13
Process 0: received rank 15 from process 15
Process 0: received rank 9 from process 9
Process 0: received rank 2 from process 2
Process 0: received rank 4 from process 4
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 12 from process 12
Process 0: received rank 14 from process 14
============= Execution Results =============
Num of proc: 16 | Execution time: 0.000106 seconds
```

3. Построим графики времени выполнения и ускорения для 2-16 процессов.

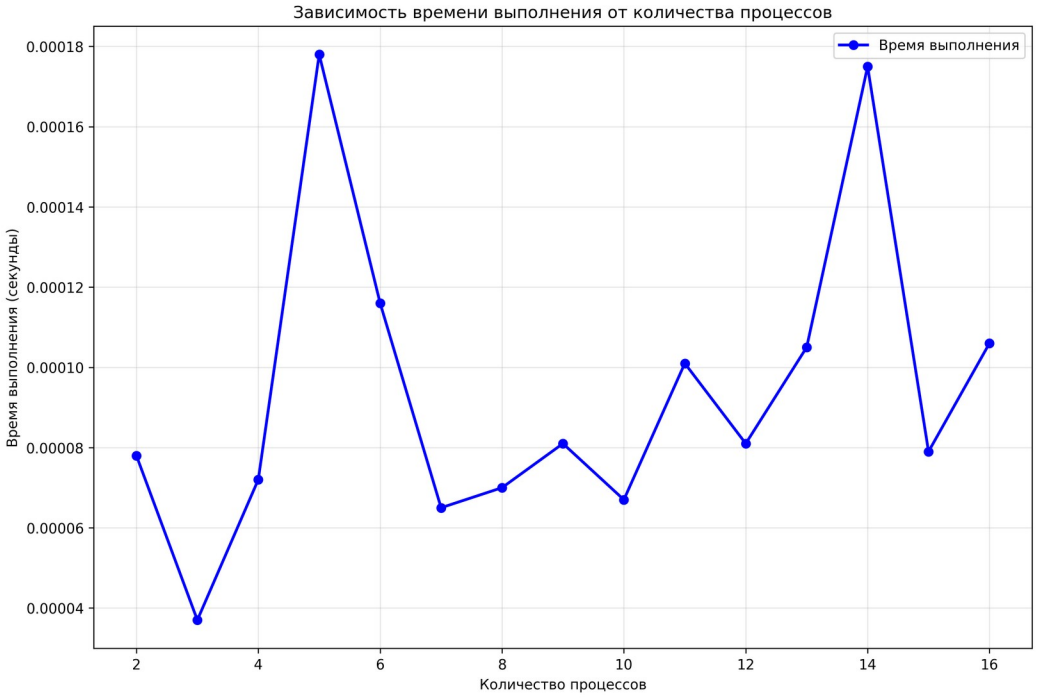

Рис.2 – график зависимости времени выполнения от числа процессов

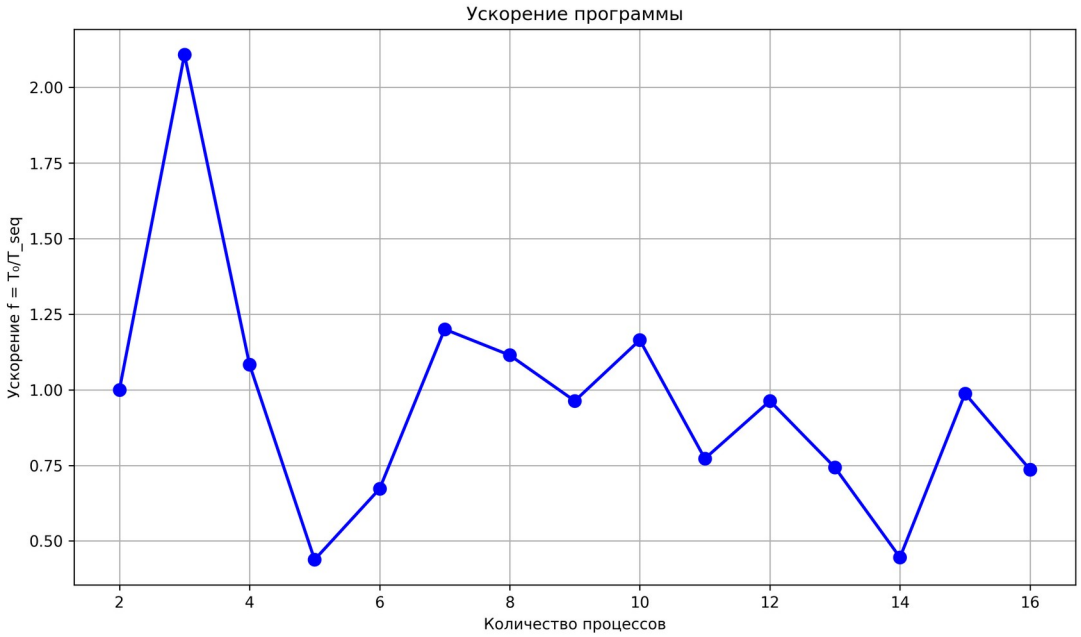

Рис.3 - график ускорения

4. Преобразуем программу, чтобы порядок вывода сообщений на экран соответствовал номеру соответствующего процесса. Для этого изменим аргументы source и tag в функции MPI_Recv на i (переменную цикла) и 0, чтобы ожидать данные от соответствующего процесса.

5. Построим схему Петри для второй программы.

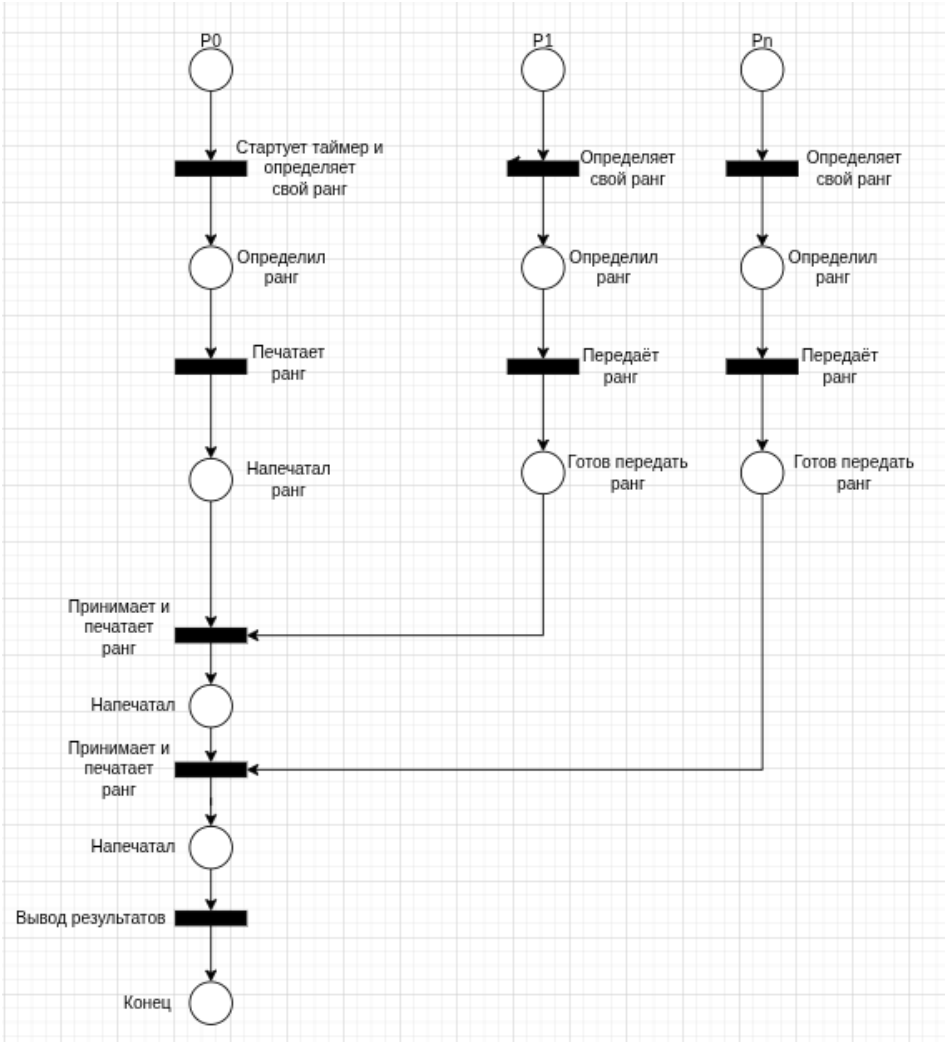

Рис.4 - сеть Петри для второй программы

6. Запустим программу и получим следующие результаты:

Таблица 2 – результаты экспериментов

N	2	4	6	8	10	12	14	16
t, сек	0.000074	0.000084	0.000040	0.000109	0.000093	0.000148	0.000097	0.000136

Результаты эксперимента:

Запуск с 2 процессами...

Process 0: my rank is 0

Process 0: received rank 1 from process 1

========== Execution Results ==========

Num of proc: 2 | Execution time: 0.000074 seconds

Запуск с 3 процессами...

Process 0: my rank is 0

Process 0: received rank 1 from process 1

```
Process 0: received rank 2 from process 2
============= Execution Results =============
Num of proc: 3 | Execution time: 0.000104 seconds
---
Запуск с 4 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
============= Execution Results =============
Num of proc: 4 | Execution time: 0.000084 seconds
---
Запуск с 5 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
============= Execution Results =============
Num of proc: 5 | Execution time: 0.000060 seconds
---
Запуск с 6 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
============= Execution Results =============
Num of proc: 6 | Execution time: 0.000040 seconds
---
Запуск с 7 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
============= Execution Results =============
Num of proc: 7 | Execution time: 0.000107 seconds
---
Запуск с 8 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
============= Execution Results =============
Num of proc: 8 | Execution time: 0.000109 seconds
---
Запуск с 9 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
```

```
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
============= Execution Results =============
Num of proc: 9 | Execution time: 0.000110 seconds
---
Запуск с 10 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
============= Execution Results =============
Num of proc: 10 | Execution time: 0.000093 seconds
---
Запуск с 11 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
============= Execution Results =============
Num of proc: 11 | Execution time: 0.000091 seconds
---
Запуск с 12 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
============= Execution Results =============
Num of proc: 12 | Execution time: 0.000148 seconds
---
Запуск с 13 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
```

Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 12 from process 12
========== Execution Results ==========
Num of proc: 13 | Execution time: 0.000111 seconds

Запуск с 14 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 12 from process 12
Process 0: received rank 13 from process 13
========== Execution Results ==========
Num of proc: 14 | Execution time: 0.000097 seconds

Запуск с 15 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6
Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 12 from process 12
Process 0: received rank 13 from process 13
Process 0: received rank 14 from process 14
========== Execution Results ==========
Num of proc: 15 | Execution time: 0.000135 seconds

Запуск с 16 процессами...
Process 0: my rank is 0
Process 0: received rank 1 from process 1
Process 0: received rank 2 from process 2
Process 0: received rank 3 from process 3
Process 0: received rank 4 from process 4
Process 0: received rank 5 from process 5
Process 0: received rank 6 from process 6

Process 0: received rank 7 from process 7
Process 0: received rank 8 from process 8
Process 0: received rank 9 from process 9
Process 0: received rank 10 from process 10
Process 0: received rank 11 from process 11
Process 0: received rank 12 from process 12
Process 0: received rank 13 from process 13
Process 0: received rank 14 from process 14
Process 0: received rank 15 from process 15
========== Execution Results ==========
Num of proc: 16 | Execution time: 0.000136 seconds

8. Построим графики аналогично первой задаче.

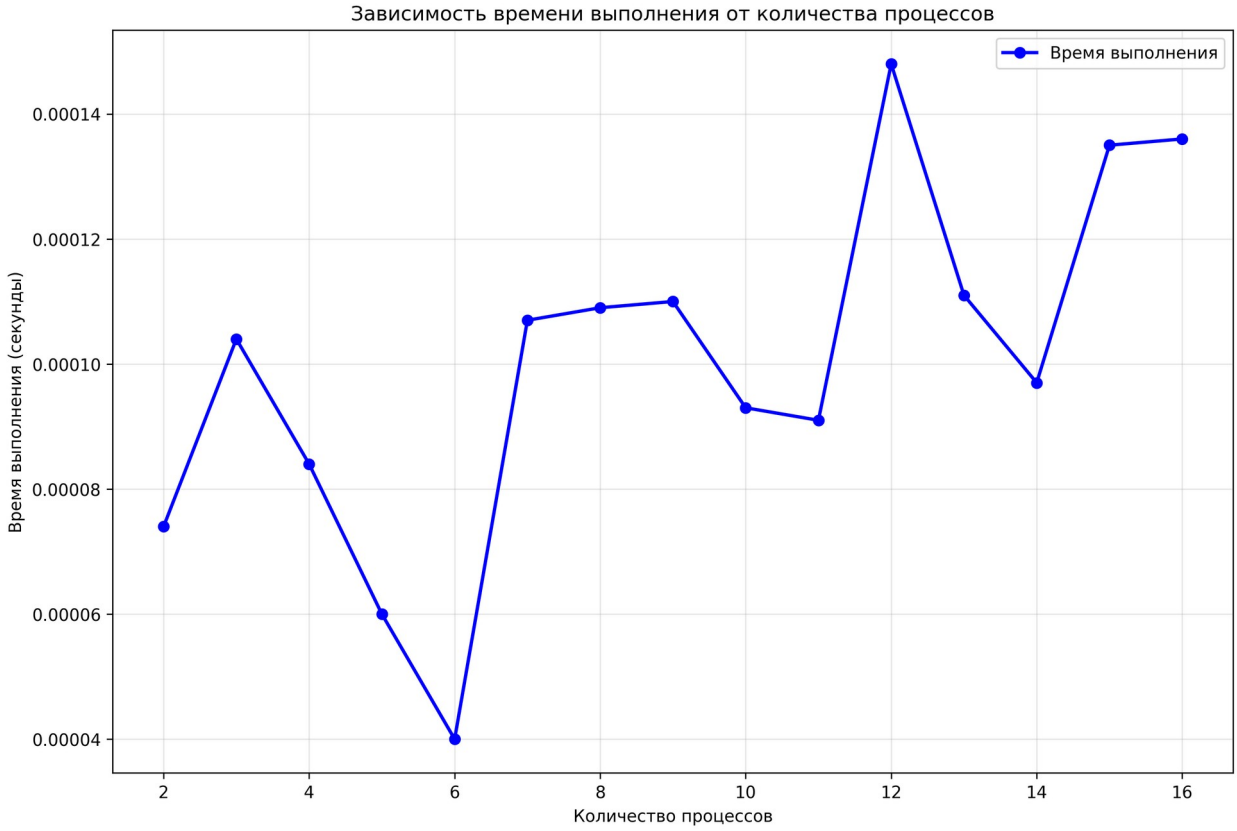

Рис.5 – график зависимости времени выполнения от числа процессов

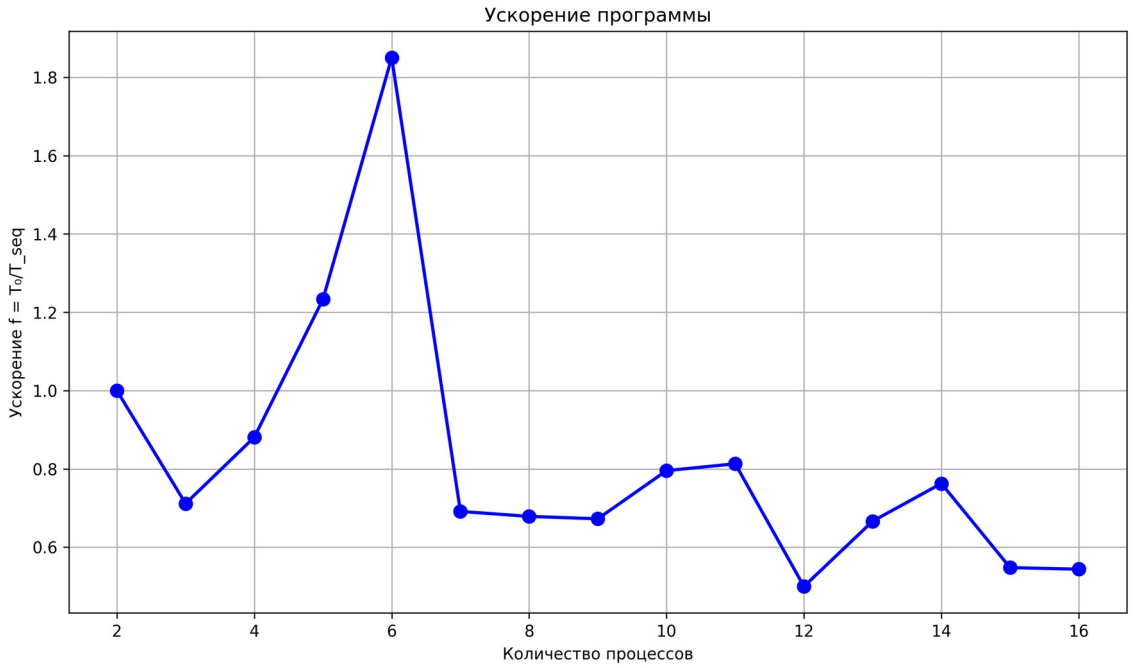

Рис.6 - график ускорения

Разработанный программный код см. в приложении А.

Выводы.

В рамках лабораторной работы создана параллельная MPI-программа. Изучены основы обмена сообщениями, включая недетерминизм при получении данных. Замеры производительности показали, что алгоритм с последовательным приемом сообщений от процессов в среднем приводит к увеличению времени работы программы на 0.0001 секунды по сравнению с непоследовательным приёмом. Параллельное выполнение не привело к ускорению программы ни в одном из случаев.

Приложение А

Исходный код программы

Название файла: task1.c

```
#include <stdio.h>
#include <mpi.h>

int main(int argc, char **argv)
{
    int rank, size;

    MPI_Status status;

    MPI_Init(&argc, &argv);

    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (rank == 0) {
        double start_time, end_time;
        start_time = MPI_Wtime();

        printf("Process 0: my rank is %d\n", rank);

        for (int i = 1; i < size; ++i) {
            int received_rank;
            MPI_Recv(&received_rank,
                    1,
                    MPI_INT,
                    MPI_ANY_SOURCE,
                    MPI_ANY_TAG,
                    MPI_COMM_WORLD,
                    &status);

            printf("Process 0: received rank %d from process %d\n",
                    received_rank, status.MPI_SOURCE);
        }

        end_time = MPI_Wtime();
        double exec_time = end_time - start_time;

        puts("============= Execution Results =============");
        printf("Num of proc: %d | Execution time: %.6f seconds",
                size, exec_time);
```

```
    }
    else {
        MPI_Send(&rank, 1, MPI_INT, 0, 0, MPI_COMM_WORLD);
    }

    MPI_Finalize();
    return 0;
}
```

Название файла: task2.c

```
#include <stdio.h>
#include <mpi.h>

int main(int argc, char **argv)
{
    int rank, size;
    MPI_Status status;

    MPI_Init(&argc, &argv);

    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &size);

    if (rank == 0) {
        double start_time, end_time;
        start_time = MPI_Wtime();

        printf("Process 0: my rank is %d\n", rank);

        for (int i = 1; i < size; i++) {
            int received_rank;
            MPI_Recv(&received_rank,
                    1,
                    MPI_INT,
                    i,
                    0,
                    MPI_COMM_WORLD,
                    &status);

            printf("Process 0: received rank %d from process %d\n",
                    received_rank, status.MPI_SOURCE);
        }

        end_time = MPI_Wtime();
        double exec_time = end_time - start_time;
```

```
        puts("============= Execution Results =============");
        printf("Num of proc: %d | Execution time: %.6f seconds",
                size, exec_time);

    }
    else {
        MPI_Send(&rank, 1, MPI_INT, 0, 0, MPI_COMM_WORLD);
    }

    MPI_Finalize();
    return 0;
}
```