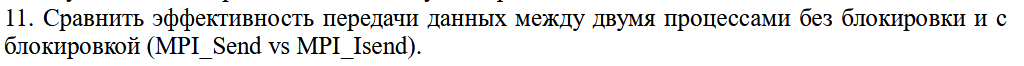
Контрольная №1

Задача на работу с Send и Isend

Боязитов Вадим, 932125

25.04.2025

## Задание 1



### Алгоритм

1. *Подготовка*

Инициализировать MPI (MPI\_Init).

Инициализация MPI и подготовка данных

Все процессы:

Инициализируется MPI (MPI\_Init).

Получают свой rank и общее количество процессов size.

Процесс 0:

Создает массив data размером ARRAY\_SIZE и заполняет его случайными числами (fill\_random\_array).

Синхронизация перед тестами

2. *Запуск блокирующей передачи*

Процесс 0:

Засекает время (MPI\_Wtime).

Распределяет данные:

В цикле отправляет каждому процессу его часть массива (MPI\_Send).

Обрабатывает свою часть:

Копирует свою часть данных в local\_data.

Вычисляет частичную сумму (compute\_partial\_sum).

Собирает результаты:

Принимает частичные суммы от всех процессов (MPI\_Recv).

Суммирует их в total\_sum.

Выводит результат и время выполнения.

Остальные процессы:

Получают данные (MPI\_Recv).

Вычисляют частичную сумму.

Отправляют результат процессу 0 (MPI\_Send).

3. *Запуск неблокирующей передачи*

Процесс 0:

Засекает время (MPI\_Wtime).

Распределяет данные асинхронно:

Отправляет части массива с помощью MPI\_Isend (неблокирующая отправка).

Обрабатывает свою часть:

Копирует свою часть в local\_data.

Вычисляет частичную сумму.

Ждет завершения всех отправок (MPI\_Waitall).

Собирает результаты синхронно (MPI\_Recv).

Выводит результат и время выполнения.

Остальные процессы:

Получают данные (MPI\_Recv).

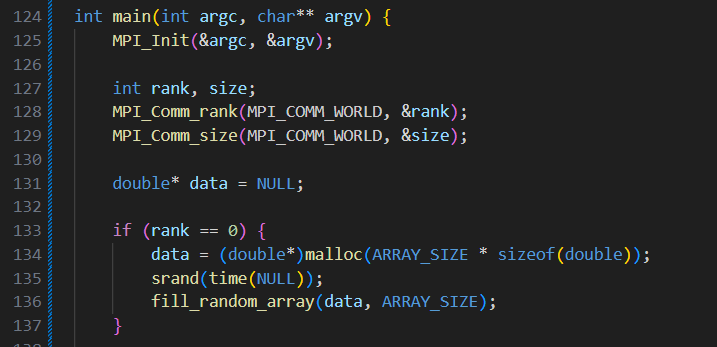
Вычисляют частичную сумму.

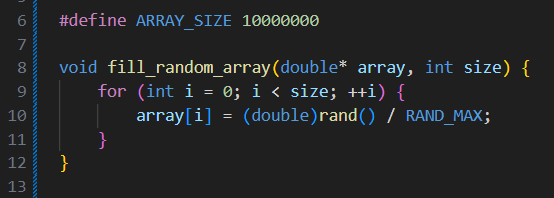
Отправляют результат процессу 0 (MPI\_Send).

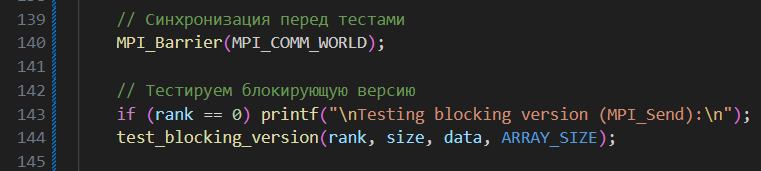
### Программная реализация

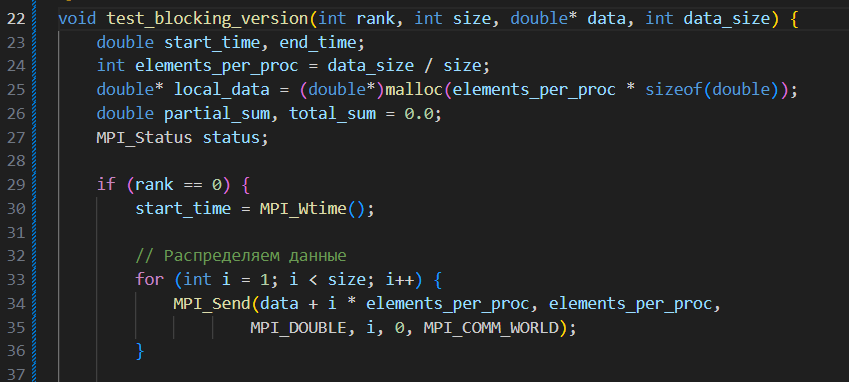
Окружение Linix. Язык C.

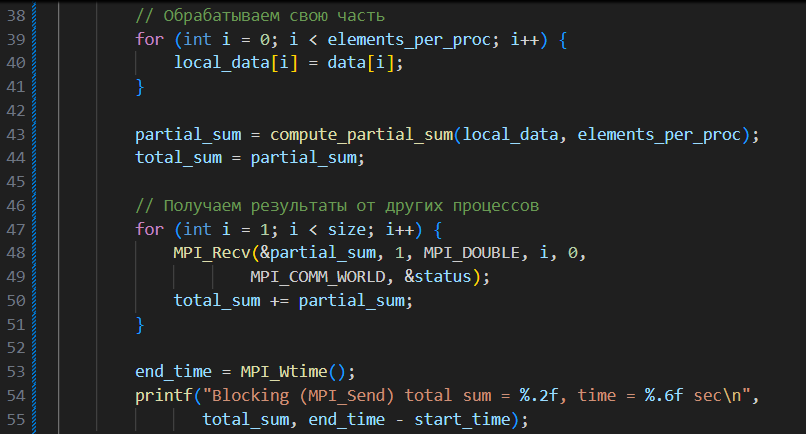
https://github.com/katet3/tsu\_parallel\_programming/blob/main/lab\_1/lab\_1.c

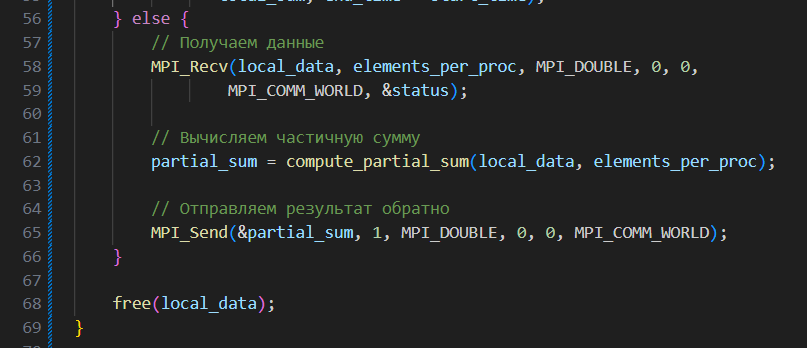


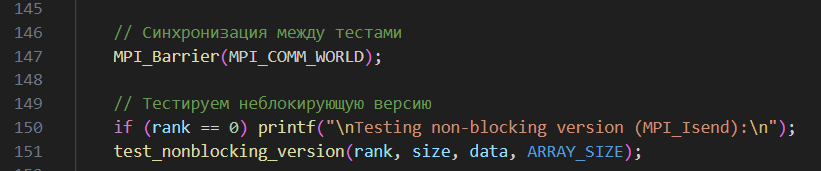


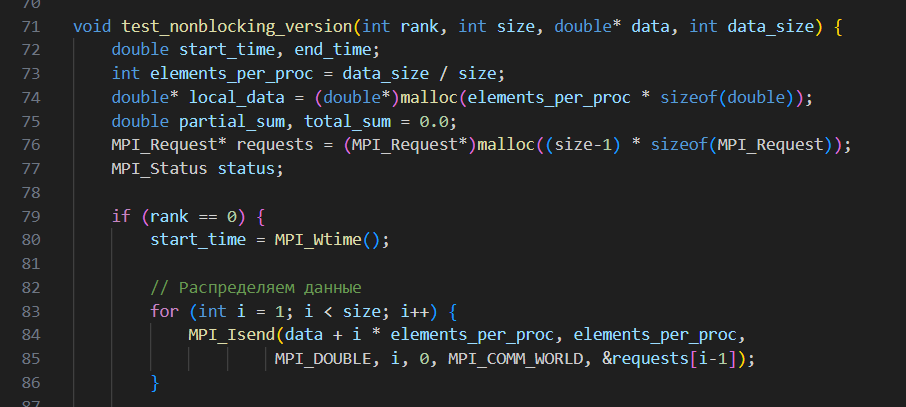


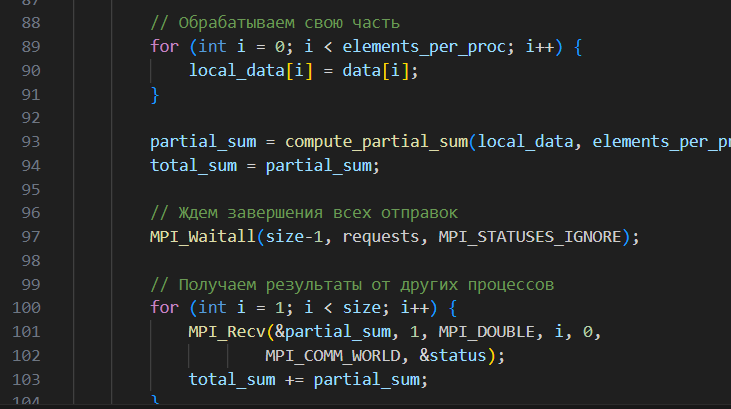


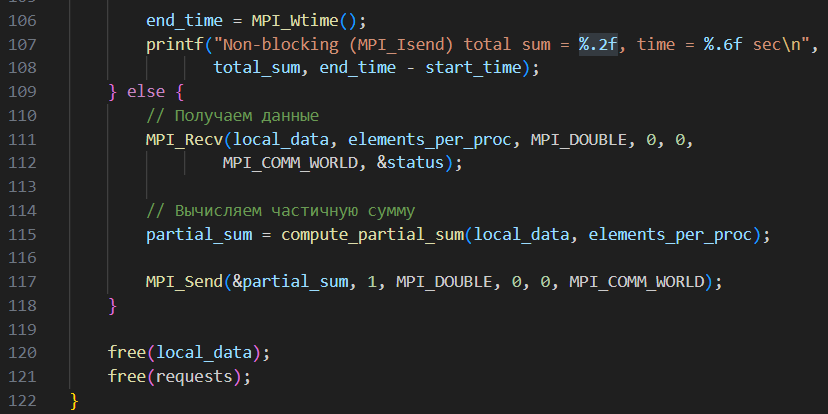


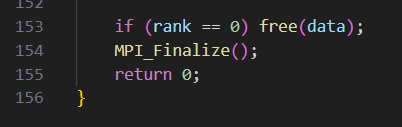




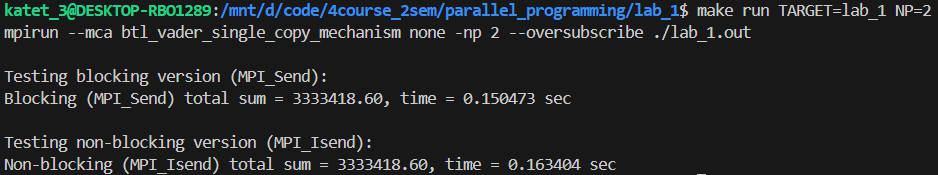


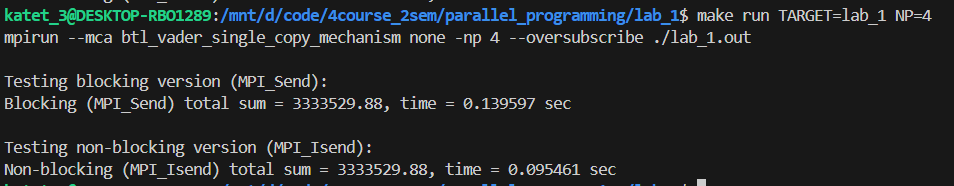


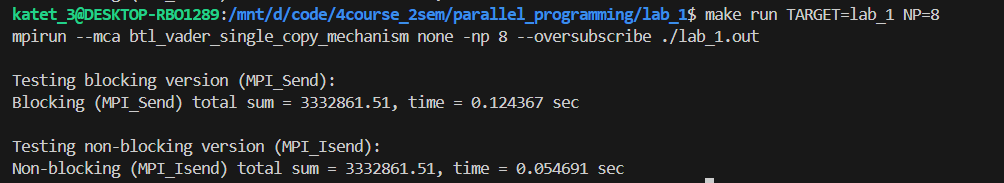




### Тестирование







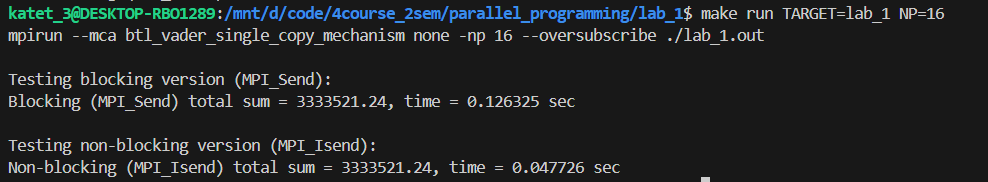
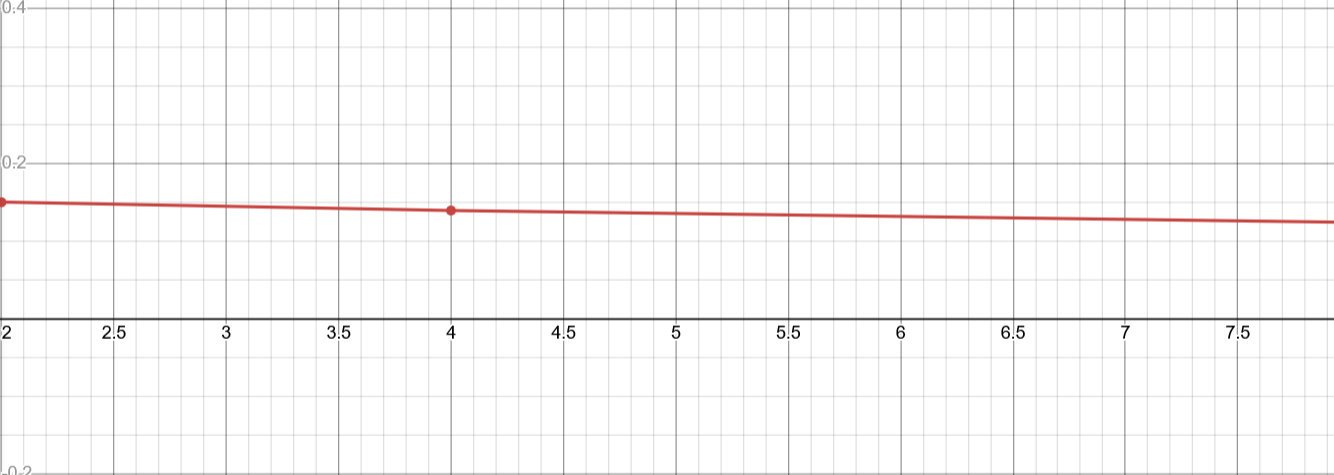


График для Isend, x-кол-во процессов, y - время



График для Send, x-кол-во процессов, y - время



### Вывод

При большом кол-ве процессов при использовании Send для отправки, большая часть времени будет заниматься на ввод-вывод. При Isend нагрузка будет размазываться.