МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное бюджетное образовательное учреждение высшего образования  
 «Кемеровский государственный университет»**

**Институт фундаментальных наук**

**ДОМАШНЯЯ РАБОТА №2**

**ПО ДИСЦИПЛИНЕ**

**“Технологии параллельных вычислений”**

**Задача № 1**

студента 3 курса

**Сулима Роман Иванович**

Направление 02.03.02 – Фундаментальная информатика и информационные технологии

Преподаватель:

к-т физ.-мат.наук, доцент

С.В. Стуколов

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Работа защищена:

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_201\_г.

с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_

Кемерово 2021

СОДЕРЖАНИЕ

[1. Постановка задачи 2](#_Toc335561691)

[2. Описание используемых функций 2](#_Toc335561692)

[3. Описание программы 2](#_Toc335561693)

[4. Реализация 2](#_Toc335561694)

[Заключение 4](#_Toc335561695)

[Литература 4](#_Toc335561696)

# 1. Постановка задачи

Написать программу, используя блокирующие коммуникационные функции (MPI\_Send, MPI\_Recv), реализующую алгоритм передачи данных по кольцу.

# 2. Описание используемых функций

MPI\_Send() – Передача сообщения другим потокам.

MPI\_Recv() – Принимает сообщение из другого потока.

# 3. Описание программы

Программа выполняет передачу сообщения по кольцу. Она отправляет сообщение с 0 потока на следующий и завершает выполнение, когда сообщение обратно возвращается на 0 поток.

# 4. Реализация

Программный код:

#include "mpi.h"

#include "stdlib.h"

#include <stdio.h>

int main(int argc, char \*argv[])

{

    int n = 5;

    int i = 0;

    int rank;

    int size;

    MPI\_Status stat;

    MPI\_Init(&argc, &argv);

    MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank);

    MPI\_Comm\_size(MPI\_COMM\_WORLD, &size);

    if (rank == 0)

    {

        MPI\_Send(&n, 1, MPI\_INT, 1, 777, MPI\_COMM\_WORLD);

        MPI\_Recv(&n, 1, MPI\_INT, (size - 1), 777, MPI\_COMM\_WORLD, &stat);

    } else if (rank == (size - 1)) {

        MPI\_Recv(&n, 1, MPI\_INT, (size - 2), 777, MPI\_COMM\_WORLD, &stat);

        MPI\_Send(&n, 1, MPI\_INT, 0, 777, MPI\_COMM\_WORLD);

    }

    else

    {

        MPI\_Recv(&n, 1, MPI\_INT, rank - 1, 777, MPI\_COMM\_WORLD, &stat);

        MPI\_Send(&n, 1, MPI\_INT, rank + 1, 777, MPI\_COMM\_WORLD);

    }

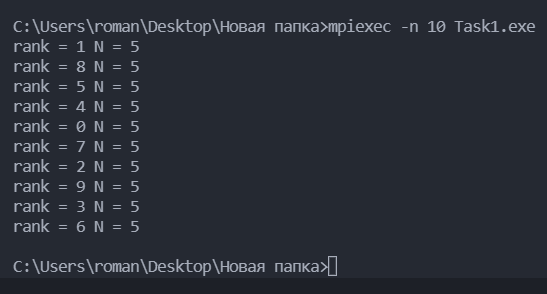
    printf("rank = %d N = %d \n", rank, n);

    MPI\_Finalize();

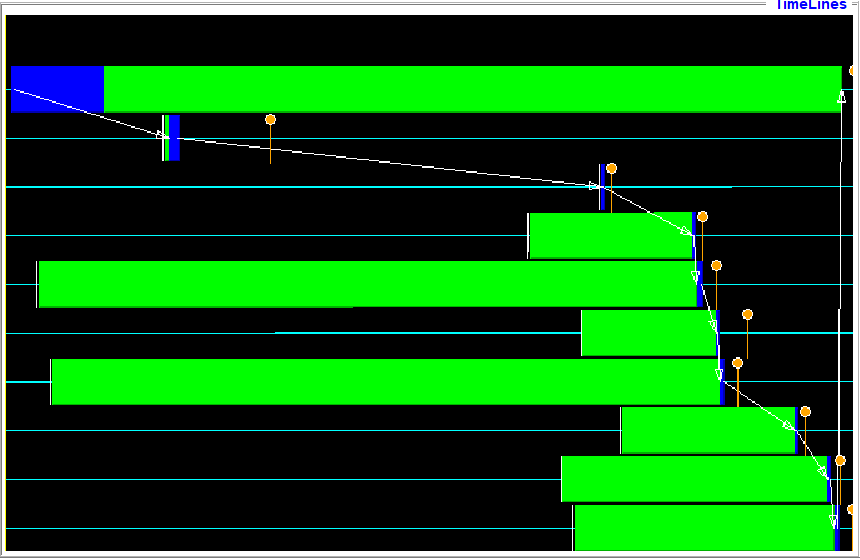
    return 0;

}

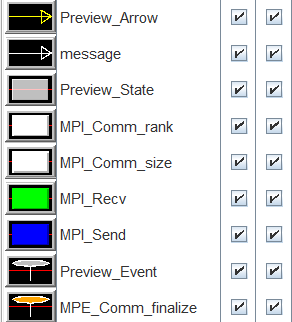
Пример реализации:



Трасса:



Условные обозначения:



# Заключение

Изучен способ передачи переменной в разные потоки, а так же изучена реализация алгоритма передачи по кольцу.

# Литература