

Московский Государственный Университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики

Отчет по учебному курсу «Распределенные системы»

Бирюков Андрей Матвеевич
427 группа

2021 год

Постановка задачи

Необходимо добавить в реализацию прошлогодней задачи возможность ее продолжения в случае сбоя одного процесса.

Реализация параллельной сортировки MergeSort.

Реализация алгоритма

1. Запуск:

Для выполнения этого задания необходимо дополнение mpi-ext. Для этого был скомпилирован mpi из исходников для MacOS.

Запуск осуществлялся так:

```
mpicc mergeMPI.c
```

```
mpirun -np 4 --with-ft ulfm --oversubscribe --mca shmem posix ./a.out N
```

N – число элементов массива

2. Программа:

1. Сортировка слиянием реализована методом деления массива чисел на N частей и рассылка всех частей процессам через MPI_Scatter.
Для каждого процесса создается файл, в который он записывает полученный неотсортированный массив сразу после того, как получает его.
Далее части сортируются и сохраняются в этот же файл процесса.
2. Допустим, процесс с номером KILL_PROC останавливает свою работу (в нашем случае это искусственное создание ситуации через sigkill) во время сортировки и отсортированный массив не был сохранен в файл.
3. В качестве процесса, который будет «на подхвате» и продолжит вычисления – будет координатор. Если бы процесс, который мы убиваем, совпал бы с номером координатора – в БОЛЬШИХ масштабах программы можно было бы применить алгоритм Задиры для выбора нового координатора, реализованный ранее в первом задании.
4. С помощью функций MPI_Comm_create_errhandler и MPI_Comm_set_errhandler добавлен обработчик ошибок verbose_errhandler
5. После того как убивается один процесс, управление переходит в функцию-обработчик ошибок.
6. В обработчике ошибок на базе старого коммуникатора создается новый, не включающий в себя убитый процесс.
7. После создания нового коммуникатора, каждый процесс получает новый номер и, возможно, работает с другим файлом, но это не влияет на результат работы программы.
8. Файл убитого процесса считывается и сортируется процессом-координатором и сохраняется в новый файл со специальным названием.
9. В основной программе MPI_Gather собирает отсортированные части массива с работающих процессов, а процесс координатор добавляет в конец массива недостающую часть отсортированных чисел из файла, созданного в обработчике.
10. В конце все отсортированные куски сливаются в один отсортированный массив.
11. Проверка сортировки происходит при помощи сравнения полученного нашей программой массива и отсортированного массива с помощью сортировки пузырьком.

Вывод

Была реализована программа, которая имеет возможность продолжить работу при сбое одного процесса при помощи расширения `mpi` – «`mpi-ext.h`»