

Московский государственный университет имени М. В. Ломоносова
Факультет вычислительной математики и кибернетики



Параллельные высокопроизводительные вычисления

Отчет по заданию №1
«Расписание сети слияния»

Бирюков Андрей Матвеевич
528 группа

30 октября 2022 года

1. Описание условия

Необходимо разработать последовательную программу вычисления:

- расписания сети объединения двух упорядоченных массивов с размерами $p1$ и $p2$.
- числа использованных компараторов
- числа тактов, необходимых для её выполнения на $p = p1 + p2$ процессорах.

Число тактов сортировки при параллельной обработке не должно превышать числа тактов, затрачиваемых соответствующей сетью четного-нечетного слияния Бетчера.

Параметры командной строки запуска: $p1, p2$. ($p1 \geq 1, p2 \geq 0$ – числа элементов в упорядоченных объединяемых массивах, элементы которых расположены в строках с номерами $[0...p1-1]$ и $[p1...p1 + p2-1]$ соответственно)

Формат команды запуска: **bjoin p1 p2**

Требуется:

- вывести в файл стандартного вывода расписание и его характеристики в представленном далее формате;
- обеспечить возможность вычисления сети сортировки для числа элементов $1 \leq p1 + p2 \leq 10000$;
- предусмотреть полную проверку правильности сети сортировки для значений числа сортируемых элементов $1 \leq p1 + p2 \leq 24$;
- представить краткий отчет, удовлетворяющий указанным далее требованиям.

Формат файла результата:

Начало файла результата

$p1\ p2\ 0$

$cu0\ cd0$

$cu1\ cd1$

...

$cun_comp-1\ cdn_comp-1$

n_comp

n_tact

Конец файла результата

($p_1, p_2, 0$ – размеры первого и второго упорядоченных массивов, число 0 su_i
 cd_i – номера строк, соединяемых i -м компаратором сравнения –
перестановки, n_comp – общее число компараторов, n_tact – общее число
тактов сети слияния)

2. Описание метода решения

По условию задания требуется реализовать слияние заранее отсортированных массивов. В результате работы программы нужно получить последовательность компараторов, при помощи которой можно будет получить отсортированный массив размера $p1 + p2$.

Для этого была реализована часть алгоритма четно-нечетной сортировки Бэтчера. Согласно алгоритму массив разделяется на 2 равные части (или же на 2 части с разницей в 1 элемент в случае нечетного числа элементов). Далее следует отсортировать каждую из частей и объединить результаты с помощью сети нечетно – четного слияния Бэтчера. В сети нечетно-четного слияния отдельно объединяются и сортируются элементы массивов с нечетными номерами и отдельно – с четными. Далее происходит слияние результатов сортировки массивов из элементов с четными и нечетными номерами. В рамках задания необходимо было реализовать последний этап слияния двух упорядоченных массивов.

Реализация написана на языке C++.

Компиляция программы осуществляется командой:

g++ -std=c++11 -o bjoin bjoin.cpp

Запуск программы:

./bjoin p1 p2 -t(--test) где:

- $p1$ – размер первого массива
- $p2$ – размер второго массива
- $-t$ ($--test$) – флаг, который активирует тесты

3. Описание метода проверки

Тестирование проводилось при помощи 0-1 принципа.

0-1 принцип: если сеть сортирует все последовательности из нулей и единиц, то сеть является сортирующей.

Была реализована функция, которая осуществляет перебор всевозможных упорядоченных комбинаций массивов из 0 и 1, длины которых удовлетворяют ограничению: $1 \leq p1 + p2 \leq 24$.

При работе функции в стандартный поток вывода печатаются все генерируемые массивы и статус проверки их отсортированности:

- TEST SUCCEEDED, если массив отсортирован успешно
- TEST FAILED, если массив отсортирован некорректно. В данном случае программа завершает свою работу.

Если все тесты прошли успешно, то в конце будет напечатано «All TESTS PASSED».

Приложение1:

Исходный текст программы в отдельном с++ файле.