Алгоритмы для работы с большими объемами данных, практика 2

Требуется реализовать алгоритм сортировки чисел во внешней памяти с помощью сортировки слияниями.

Ваше решение должно состоять из следующих частей:

1. код программы, выполняющей сортировку;
2. реализация сортировки с помощью STXXL;
3. отчет, содержащий исследование вашего алгоритма.

Программа

В качестве аргументов командной строки программа должна принимать:

- имя входного файла;
- имя выходного файла;
- путь до временной директории;
- объем доступной оперативной памяти M в байтах;
- размер блока B в байтах.

Во входном файле будут подряд записаны беззнаковые 64-битные числа в бинарном little-endian формате.

Ваша реализация должна поддерживать сортировку файлов любого разумного размера и позволять указывать любые разумные значения для M и B. Можно предполагать, что оба числа кратны 4КБ, верно соотношение $M \geq 32B$, а также вашей программе будет дан запас памяти поверх M на накладные расходы (как минимум 16МБ).

В архиве с заданием есть пример входных и выходных данных, а также пример программы, выполняющей сортирвку во внутренней памяти. Требуется в точности следовать формату входных данных и аргументов командной строки. Все промежуточные файлы должны создаваться во временной директории.

Отчет

В отчете должно содержаться следующее:

1. Краткая характеристика вашего железа:
   - Объем RAM;
   - Тип жесткого диска, измеренное время seek и скорость последовательного чтения.
2. Исследование работы вашего алгоритма на файлах разного размера и при разном объеме доступной оперативной памяти.

   Стоит брать файлы размером 2ГБ и 20ГБ (последнее значение можно немного уменьшить, если у вас не хватает места на диске). Дальше рекомендуется действовать следующим образом: зафиксировать размер блока, разумный для вашего диска (например, 256КБ если у вас обычный SSD) и варьировать размер оперативной памяти, начиная с 100 размеров блока и заканчивая значениями порядка 1-2ГБ.

   Для выполненных замеров надо вычислить, в какой пропорции затрачиваемое время делится между операциями с CPU и I/O. В исследовании хочется увидеть как время работы вашего алгоритма зависит от объема доступной памяти, как оно соотносится с предполагаемым в теории временем на чтения/записи данных.

   Тестирование рекомендуется проводить на операционной системе Linux с использование cgroup-ы, ограниченной по памяти. В архиве будет пример кода для создания cgroup-ы и запуска в ней программы.
3. Сравнения времени работы вашего алгоритма с реализацией на основе STXXL

   В архиве у вас будет дан пример программы с использованием библиотеки STXXL. Данный пример требуется доработать, чтобы он производил сортировку чисел; после этого надо произвести сравнение вашей реализации с сортировкой с помощью STXXL.

Документация: `https://stxxl.org/`

Архив

Архив с заданием состоит из следующих частей:

1. `example/` — директория с примером решения и входными/выходными данными.
2. `stxxl/` — пример программы с использованием STXXL, которая проверяет сортированность чисел во входном файле.
3. `create_cgroup.sh` — скрипт для создания cgroup-ы.

Критерии оценки

В первую очередь решение будет оцениваться по корректности работы алгоритма. Иными словами, неправильно работающие решения или решения использующие больше оперативной памяти, чем задано засчитываться не будут (мы постараемся доделать тестовую систему для данного задания, чтобы вы могли проверить корректность вашего решения в боевых условиях).

Во вторую очередь решение будет оцениваться по времени его работы и оптимальности реализации в общем случае. Будет проводиться сравнение с сортировкой во внутренней памяти, с STXXL и с другими реализациями слушателей курса.

Также, важная часть решения — это отчет; не менее 4 баллов за задачу будет зависеть от полноты отчета и проведенного исследования.