External Sorting

Read the chapter on external sorting from Raghu Ramakrishnan's book.

The input file format is as follows:

- Number of available buffer pages (K)
- Number of runs to merge together (M)
- Number of records per page (R)
- Number of records in the data file (N)
- List of records. Assume that each record is just a non-negative integer. There could be duplicates in the data file.
- Query: iteration number, page number

Your datafile have P=ceiling(N/R) pages.

In iteration 0, you should create sorted runs of size K. Total number of runs at the end of iteration 0 will be ceiling(P/K).

In each subsequent iteration, you should merge M runs at a time.

After each iteration, you will be writing the whole data file to the secondary storage.

Output:

You are supposed to write contents of a particular data page at the end of a particular iteration.

Please note that numbering of pages as well as iterations starts from zero.

Example input:

We have 6 buffer pages.

We have to perform three-way merge sort.

Two records fit per page.

The data file has 105 records.

We have report the status of external merge sort after iteration 1 and print the contents of page 1.

Output:

The target page contains keys 72 and 73.

Explanation: Iteration 0 will create sorted runs of size 6 pages each.

In iteration 1, we will combine 3 runs at a time.

Note on code submission:

Make sure that all your code fits into a single file <roll number>ESort.cpp

How your code should compile?

g++ <roll number>ESort.cpp

How your code should run?

a.out < input_file_name

Your code should write the output on the stdout.