Quiz 11 – Solution

Consider natural-joining two relations R(A, B) and S(B, C) using the simple sort-based join algorithm (where input relations are completely sorted first).
Suppose M = 100 pages, B(R) = 5,000 blocks, and B(S) = 20,000 blocks. Assume all memory is used in the join process (including sorting). Also assume that the situation described in class where there are too many join tuples does **not** occur here.

1. [6 points] Describe the steps in sorting R and S (i.e., for each pass, how many runs are generated at each pass and the size of each run).

   For R:
   Pass 1: Split R into runs of size M -> get 50 sorted lists of size 100 [1]
   Pass 2: Merge 50 runs -> get 1 sorted list of size 5,000 [1]

   For S:
   Pass 1: Split S into runs of size M -> get 200 sorted lists of size 100 [1]
   Pass 2: Merge M-1 = 99 runs (one output buffer is needed) -> get 1 sorted list of size 9,900
   Merge another M-1 = 99 runs -> get 1 sorted list of size 9,900
   Merge the left 2 runs -> get 1 sorted list of size 200 [2]
   Pass 3: Merge 3 runs -> get 1 sorted list of size 20,000 [1]

2. [2 points] What is the cost (# of block I/O's) of the above sorting process?

   For R: 2B(R) * 2 = 20,000 [1]
   For S: 2B(S) * 3 = 120,000 [1]
   Total: 140,000

3. [2 points] What is the **total** cost of the join algorithm (ignore the cost of writing the join results)?

   Cost of sorting: 140,000
   Cost of joining: B(R) + B(S) = 25,000
   Total: 165,000