

13.2. We will refer to the tuples (kangaroo, 17) through (baboon, 12) using tuple numbers t_1 through t_{12} . We refer to the j -th run used by the i -th pass, as r_{ij} . The initial sorted runs have three blocks each. They are:

$$\begin{aligned} r_{11} &= \{t_3, t_1, t_2\} \\ r_{12} &= \{t_6, t_5, t_4\} \\ r_{13} &= \{t_9, t_7, t_8\} \\ r_{14} &= \{t_{12}, t_{11}, t_{10}\} \end{aligned}$$

Each pass merges two runs. Therefore the runs after the end of the first pass are:

$$\begin{aligned} r_{21} &= \{t_3, t_1, t_6, t_5, t_2, t_4\} \\ r_{22} &= \{t_{12}, t_{11}, t_{10}, t_9, t_7, t_8\}. \end{aligned}$$

Another pass merges these two runs into the final run r_{31} with all the records.

13.5. We can store the entire smaller relation in memory, read the larger relation block by block and perform nested loop join using the larger one as the outer relation. The number of I/O operations is equal to $b_r + b_s$, and memory requirement is $\min(b_r, b_s) + 1$ pages.

Extra exercise. After preliminary sorting, we have $600000 / 150 = 4000$ runs. After the first merging step, $\lceil 4000 / 149 \rceil = 27$ runs left. Another merging step produces the sorted file.

Hence, the total cost is $2 \cdot 600000 \cdot 3 = 3.6$ million I/Os.