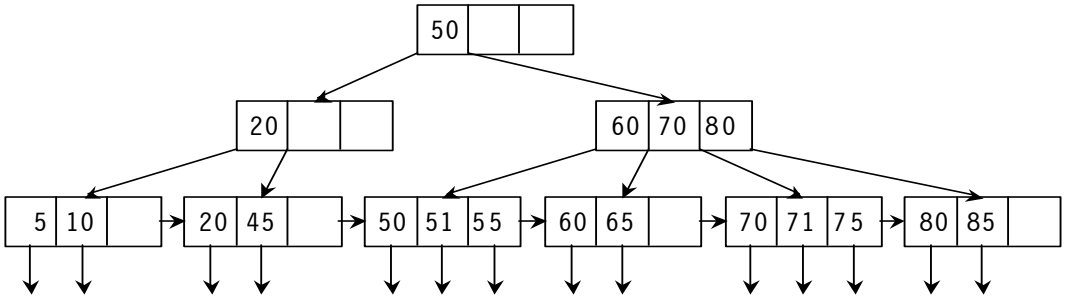
CPS 196.3 Fall 2002
Homework #4
Assigned: Tuesday, November 19
Due: Tuesday, November 26

Problem 1.

For each of the following modifications, show the result B^+-tree obtained by applying the modification to the B^+-tree shown below. Suppose that the maximum fan-out is 4. (Always start with the B^+-tree shown below; do not apply the modifications to the result of previous modifications.)



(a) Insert 21.
(b) Delete 50.
(c) Insert 79.
(d) Delete 10.

Problem 2.

A table $R(\underline{K}, A, \ldots)$ with 100,000 rows is stored in 10,000 disk blocks. The rows are sorted by K, but not by A. There is a dense, secondary B^+-tree index on $R(A)$, which has 3 levels and 500 leaves.

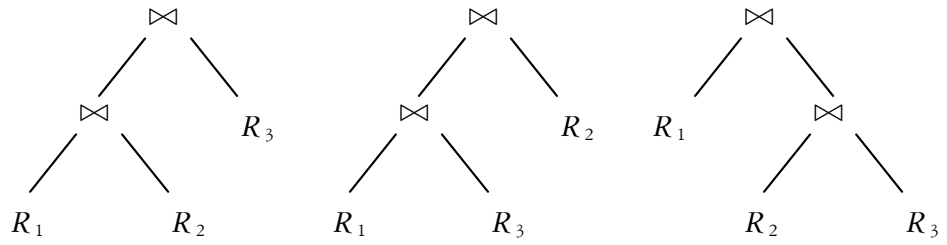
Suppose we want to sort R by A. We have 101 memory blocks at our disposal. Method 1 performs an external-memory merge sort using all memory available. Method 2 takes advantage of the fact that the values of A are already sorted in the B^+-tree index on $R(A)$: It simply scans the leaves of the index to retrieve and output R rows in order.

How many disk I/O's do these two methods require? Which one is the winner?

Problem 3.

How many possible plans are there for an n-way join query $R_1 \bowtie R_2 \bowtie \ldots \bowtie R_n$, if we use only one type of asymmetric binary join operator in our plans? Your answer should be a closed-form or recurrence formula in terms of n. Also, compute your answer for $n = 7$.

Remember to consider all bushy plans—not just left-deep ones. For example, three possible plans for $n = 3$ are shown below. There are a total of 12 plans for $n = 3$.



Problem 4.

Consider tables $R(A, B, C)$, $S(C, D)$, and $T(D, E)$. Transform the following query into an equivalent query that:

- Contains no cross products;
- Performs projections and selections as early as possible.

(a) $\pi_{R.B,\, S.D,\, T.E}\, \sigma_{(R.A=10) \text{ and } (R.C = S.C) \text{ and } (S.D = T.D) \text{ and } (R.A > T.E)}\, (R \times S \times T)$

Suppose we have the following statistics:

- $| R | = 1{,}000$; $| \pi_A R | = 1{,}000$; $| \pi_B R | = 100$; $| \pi_C R | = 500$;
- $| S | = 5{,}000$; $| \pi_C S | = 300$; $| \pi_D S | = 10$;
- $| T | = 4{,}000$; $| \pi_D T | = 4{,}000$; $| \pi_E T | = 1{,}500$.

Estimate the number of the tuples returned by the following queries:

(b) $\sigma_{A=10}\, R$
(c) $\sigma_{A=10 \text{ and } B = \text{“Bart”}}\, R$
(d) $\sigma_{A=10 \text{ or } B = \text{“Bart”}}\, R$
(e) $R \bowtie S$
(f) $R \bowtie S \bowtie T$

For the following question, further suppose that:

- Each disk/memory block can hold up to 10 tuples;
- All tables are stored compactly on disk (10 tuples per block) in no particular order;
- No indexes are available;
- 11 memory blocks are available for query processing.

(g) What is the best execution plan (in terms of number of I/O's performed) you can come up with for the query $\sigma_{R.B = \text{“Bart”} \text{ and } S.D = 100}\, (R \bowtie S)$? Describe your plan and show the calculation of its I/O cost.