Handing In

Upload your homework to Gradescope.
Please write your Banner ID on your submission. Do not write your name on the submission.

Warmup #1

Explain why the time complexity of B-tree operations (insertion, deletion, and search) is $O(\log_B(N))$, where $N$ is the number of records and $B$ is the branching factor (the number of pointers).

Warmup #2

Explain the steps you would take when an insertion into a B-tree leads to a leaf node overflowing. Why can’t the inserted value fit in the node?

Warmup #3

What are the differences between B+-tree and B-tree? What are the advantages of a B-tree over a B+-tree? B+-tree over a B-tree?

Warmup #4

When is it preferable to use a dense index rather than a sparse index? Explain your answer.

Problem 5 (To Be Graded)

Construct a B+-tree for the following set of key values: (4, 5, 6, 7, 10, 12, 14, 19, 20, 21, 23)

Assume that the tree is initially empty and values are added in ascending order. Construct B+-trees for the cases where the number of pointers that will fit one node is as follows:

1. Four
2. Five
3. Seven

Using the B+-tree created in part 1, delete (in order) values 4, 5 and 6. Show each step along the way.