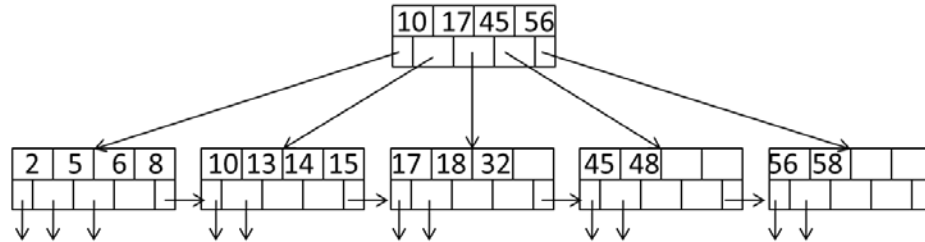


Homework #4**Due: November 10, Friday****100 points**

1. [40 points] Consider the following B+tree for the search key “age”. Suppose the degree d of the tree = 2, that is, each node (except for root) must have at least two keys and at most 4 keys.



- Describe the process of finding keys for the query condition “age ≥ 10 and age $\neq 15$ ”. How many blocks I/O’s are needed for the process?
 - Draw the updated B+tree after first deleting 45 and then deleting 18 from the tree.
 - Draw the updated tree after first inserting 7 and then inserting 11 into the tree obtained in part b.
2. [60 points] Consider natural-joining tables $R(a, b)$ and $S(a, c)$. Suppose we have the following scenario.
- R is a clustered relation with 500 blocks and 5,000 tuples
 - S is a clustered relation with 10,000 blocks and 100,000 tuples
 - S has a clustered index on the join attribute a
 - $V(S, a) = 200$ (recall that $V(S, a)$ is the number of distinct values of a in S)
 - 100 pages available in main memory for the join
 - Assume the output of join is given to the next operator in the query execution plan (instead of writing to the disk) and hence the cost of writing the output is ignored.

Describe the steps (including input, output at each step, and their sizes) in each of the following join algorithms. What is the total number of block I/O’s needed for each algorithm? Which algorithm is the most efficient?

- Nested-loop join with R as the outer relation
- Nested-loop join with S as the outer relation
- Sort-merge join
- Simple sort-based join
- Partitioned-hash join
- Index join (ignore the cost of index lookup)