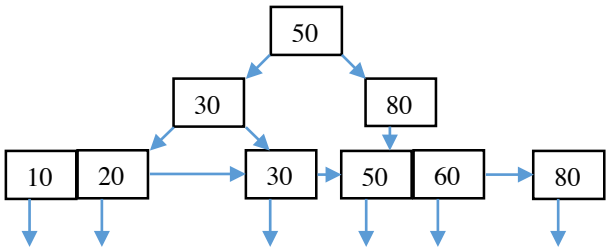
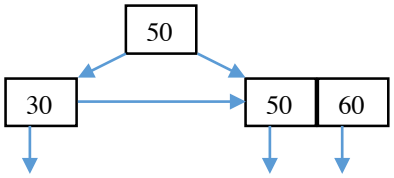
Jonathan Woong
804205763
CS 143 – LAB 1A

Homework 4

2. Show the final B+tree structure after we insert 60, 20, and 80 into Figure (a) in the given order.



Show the final B+tree structure after we delete 20, 10, and 70 from Figure (b) in the given order.



3. Consider a B+tree that indexes 300 records. Assume that n = 5 for this B+tree (i.e., each node has at most 5 pointers), what is the minimum and maximum height (depth) of the tree? (A tree with only the root node has a height of 1.)

$$\frac{300}{4} = 75\ leaf\ nodes\ at\ level\ 1$$

$$\frac{75}{5} = 15\ nodes\ at\ level\ 2$$

$$\frac{15}{5} = 3\ nodes\ at\ level\ 3$$

$$1\ node\ at\ level\ 4\ (root)$$
$$minimum\ height = 4$$

$$\frac{300}{2} = 150\ leaf\ nodes\ at\ level\ 1$$

$$\frac{150}{3} = 50\ nodes\ at\ level\ 2$$

$$\frac{50}{3} = 16\ nodes\ at\ level\ 3$$

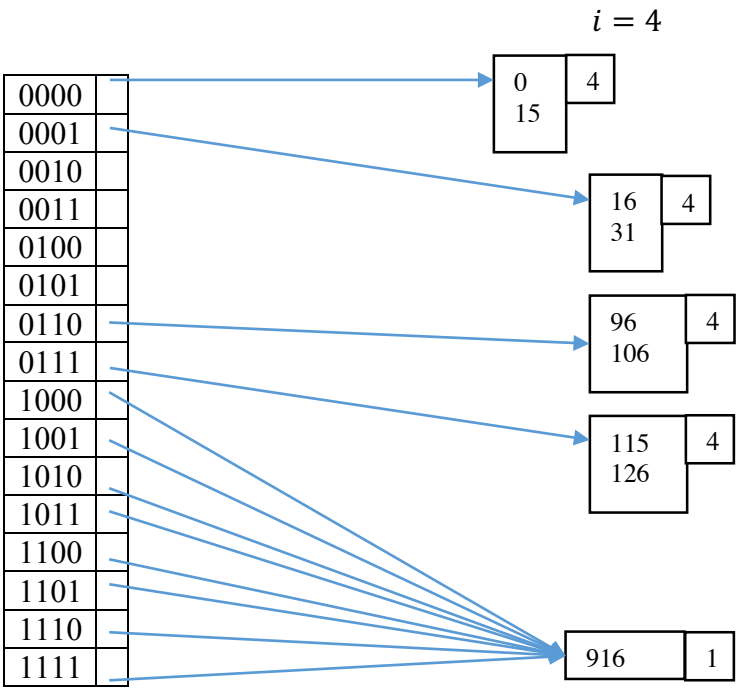
$$\frac{16}{3} = 5\ nodes\ at\ level\ 4$$

Jonathan Woong
804205763
CS 143 – LAB 1A

Homework 4

$$\frac{5}{3} = 1\ node\ at\ level\ 5\ (root)$$
$$maxmimum\ height = 5$$

3. Consider the following key values: 106, 115, 916, 0, 96, 126, 16, 15, 31
These keys are to be inserted in the above order into an (initially empty) extendible hash table. The hash function h(n) for key n is h(n) = n mod 256; that is, the hash value is the remainder when the key value is divided by 256 (28). Thus, the hash value is an 8-bit value. Each block can hold 3 data items. Draw the extendible hash table after all data items are inserted. Show the keys themselves in the buckets, not the hash value. The bucket numbers are drawn from the bits at the high order end of the hash value. Be sure to indicate i for the directory, the number of hash value bits used. Also indicate i for each bucket, the number of hash function bits that are used for that bucket.



4. Suppose you have 2 relations, R(A, B) and S(B, C), with the following characteristics:

• The size of one disk block is 1000 bytes.
• Attributes A, B are of length 10 bytes. Attribute C is of length 180 bytes.
• The tuples are not spanned across disk blocks
• |R| = 5, 000 (number of tuples of R)
• |S| = 500 (number of tuples of S)
• We have 30 blocks of memory buffer
• We use one disk block for one B+tree node

Jonathan Woong
804205763
CS 143 – LAB 1A

Homework 4

• Each pointer in a B+tree index (both a record pointer and a node pointer) uses 10 bytes.
In order to get partil credits whenyour answer is incorrect, please write down the formula that you used to obtain your answer.

1. (2 points) What is the minimum number of blocks needed to store R and S?

R: 100 S: 100

2. (2 points) We want to construct a dense B+tree on attribute B of table S by scanning the S table. What is the maximum possible branching factor n for the B+tree given the above parameters?

n: $10n + 10(n-1) \leq 1000 = 50$