CS 112: Fall 2016

Binary Trees, Huffman Coding, Hash Table Review

1. What is the worst case running time to determine the number of nodes in a binary tree, given a pointer to its root?

[✓ A. O(n)]

B. O(nlogn)

C. O(n^2)

D. None of the above

2. A binary tree has height h. (Height is number of branches from root to farthest leaf.) What is the minimum number of nodes the tree must have, if each node can have either 2 children, or no children (but not 1 child)?

A. 2h

[✓ B. 2h+1]

C. 2h-1

D. None of the above

3. For ANY binary tree, the number of external (leaf) nodes is one more than the number of internal (non leaf) nodes. True or false?

A. True

[✓ B. False]

4. What is the running time to build a Huffman tree for n symbols, given their probabilities? Assume the symbols are already sorted in increasing order of probabilities.

[✓ A. O(n)]

B. O(nlogn)

C. O(n^2)

D. None of the above

5. Given 4 symbols that occur with probabilities 0.1, 0.2, 0.3, and 0.4 respectively, what is the average code length of the resulting Huffman codes?

A. 1.5

[✓ B. 1.9]

C. 2.0

D. None of the above

6. Given 16 symbols with equal probabilities of occurrence in a document to be compressed, what would be the ratio of the length of the Huffman coded document to the original, if each instance of a symbol took up 8 bits in the original document?

A. 0.25

B. 0.5 ←

C. 0.75

D. None of the above

7. Given 6 symbols and probabilities, what is the height of the tallest Huffman tree that can result?(Height is number of branches from root to farthest leaf.)

A. 2

B. 3

C. 4

D. 5 ⬅

8. A hash table has array size (N) of 100 and load factor threshold of 0.75. How many rehashes will be done in the process of adding 200 items to this hash table? Assume the table size is doubled with every rehash:

A. 1

[✓ B. 2]

C. 3

D. 4

9. Assume an ideal hash function (distributes entries uniformly over table) is used in a hash table of initial size 10000 and load factor threshold of 1000. What is the expected running time of a search on this hash table after 1,000,000 entries are added?

[✓ A. O(1)]

B. O(n)

C. O(n^2)

D. None of the above

Each chain is of length = load factor, and the running time would be O(load factor), but since we can set the load factor threshold beforehand to some constant independent of number of items in the hash table, the running time is O(1)

10. An entry X in a hash table of size 32 has hashcode=96. Assume the entries are rehashed and the hash table doubles in size (to 64). What array index would the entry X be mapped to in the new hash table?

A. 8

B. 16

[✓ C. 32]

D. None of the above