Select the correct option for the following multiple choice questions or provide short answers accordingly:

1. Consider a hash table with chaining, of size N. Assume in each of the following scenarios the table starts empty. Further assume K pairs of <key, element> are added and K = N. Then, in which of the following situations will a find operation for a particular key have a potential worst-case runtime of O(N).
   a) The keys of all elements are powers of two and the hash function returns a unique index for each key mod N.
   b) The hash function returns a fixed number mod N, but the keys of all elements inserted are unique.
   c) Although the hashing function computes a good distribution of key values into the hash table indices, the hash function returns only even numbers.
   d) The find operation in this problem is never near O(N).

2. You have a hash table of size N = 65. You have total 10,000 elements to insert into the table (e.g., N << 10,000). The keys are all unique and are all 64 bit positive integers. You have to find a suitable hash function h(k) for key ‘k’. Which of the following hash functions will NOT work with the given parameters?
   a) h(k) = k mod N
   b) h(k) = (number of bits of k which are ‘1’) mod N
   c) h(k) = \lfloor \log_2 (k) \rfloor
   d) h(k) = \lfloor \log_2 (k^2) \rfloor

3. Consider the same setting as above. The only change is, now the keys are all ASCII strings of length 32, instead of integers. Which of the following hash function will NOT work?
   a) h(k) = (sum of the ASCII values of all the characters) mod N
   b) h(k) = (∑ k[i] * 128^i) mod N, where 0 ≤ i ≤ 31, and k[i] represent the i\textsuperscript{th} character in string k.
c) \( h(k) = (\text{ASCII value of } k[i]) / 4 \), where ‘i’ is randomly chosen, \(0 \leq i \leq 31\), \(k[i]\) represent the \(i^{th}\) character in string \(k\), and ASCII values are in the range [0, 255].

d) All the hash functions defined above will not work (in this problem).

e) All the hash functions defined above will work (in this problem).

4. You are given a hash table which allows collision. Each slot of the hash table actually keeps a linked list, and whenever a collision occurs the new element is added to the tail of the list. Suppose, you have total \(M\) elements, and the hash table is of size \(N\). Let’s assume the hash function generates hash values \(i=h(k)\) which look almost random. What is the expected running time (not the worst-case running time) for a search operation?
   a) \(O(\log M)\)
   b) \(O(\log N)\)
   c) \(O(M/N)\)
   d) \(O(1)\)

5. Explain in one sentence why each of the following is a property of a good hash function:
   a) Fast to compute
   b) Rarely causes collisions
   c) The same keys hash to the same value

6. Assume that you have an array of Strings representing words from the English language. The length of the array is \(N\). You are tasked with finding the word with the maximum and minimum frequency. Describe an algorithm where this can be done in \(O(N)\) using a hash table.

7. Assume that you have an array of unique integers. The length of the array is \(N\). You are tasked with finding two integers in this array such that their sum is equal to a provided target integer. Describe an algorithm where this can be done in \(O(N)\) using a hash table. You can assume that exactly one pair of elements in the array sums to the provided target integer.

8. The maximum number of nodes that can be present in a binary tree with depth \(n\) (Assume root is at depth 0) are:
   NOTE: You need to provide the proof of your result (i.e., you need to derive the result).

9. You have \(N\) elements equally distributed in \(K\) lists sorted in ascending order. Further the \(K\) lists are initially sorted in ascending order as well (based on the minimum element present at the start of each list). However, there is no guarantee of the relative values of the elements in the \(K\) lists below the current top of each list. Assuming no extra data structure is built:
   a) What is the tightest big-oh notation for removing the smallest element?
   b) What is the tightest big-oh notation for re-sorting the \(K\) lists once the smallest element of one of the lists is removed?
   c) What is the tightest big-oh notation for removing all \(N\) elements?
10. How many comparisons are needed by insertion sort to sort an array of 100 integers in the best possible case.
   NOTE: Provide appropriate justification on when you might be able to see the best case.
   a) 100  
   b) 4950  
   c) 99  
   d) 98

11. You run BubbleSort on a list of 10,000 items that are listed in reverse order. How many swaps will happen when running this program?
   a) 49,995,000  
   b) 50,000,000  
   c) 50,005,000  
   d) 100,000,000

12. Suppose that you have a pre-sorted linked list of size N, and you want to insert a new item into the structure in a way that keeps the list in order. What is the runtime of this operation?
   a) O(log N)  
   b) O(N)  
   c) O(1)  
   d) O(NlogN)