Name: **Batch:** **Roll No:**

CS2005:Data Structures and Algorithms, Test II, Part I, Jan. 2017 **Max Marks:15**

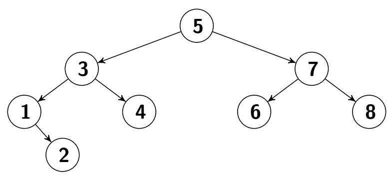
1. We have hash table of size 7 to store integers with linear probing and a hash function $h(x) = x\%7$. Fill the contents of the hash table below after inserting the keys $0, 11, 3, 7, 1, 9$ in the order. [2]

Solution:

Index	0	1	2	3	4	5	6
Key	0	7	1	3	11	9	

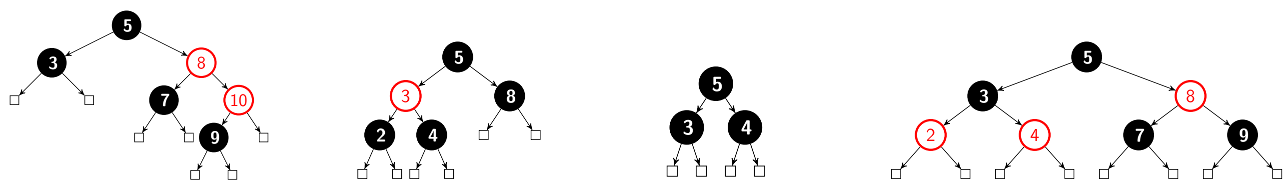
2. A Binary Search Tree contains the following values: $1, 2, 3, 4, 5, 6, 7, 8$. The tree is traversed PRE-ORDER and the values are printed out. Which of the following three sequences is a valid output. Draw the Binary Search Tree corresponding to the correct output [3]

(A) 5 3 1 2 6 4 8 5 (B) 5 3 1 2 4 7 8 6 (C) 5 3 1 2 4 7 8 6



Solution: B.

3. Below each of the Red Black Trees, specify 'YES' if it satisfies the properties of a Red-Black-Tree or 'NO' if it violates the properties. Also state the property which is violated if the answer is 'NO'. [The black colored node is black, and the other red.] [3]



Solution:
A. NO: Red node should have children colored black. Node 8 which is red has a red child.
B. YES.
C. NO: BST property. The value of the right subtree should not be less than that of the parent.
D. YES.

4. Consider the nodes of a binary search tree defined over the following node structure: [3]

```
typedef struct node {
     int data;
     struct node *parent;
     struct node *left;
     struct node *right;
};
```

Complete the definition of the following method so it returns the sum of the values contained in all of the nodes of the binary tree with root T.

Solution:

```
int sum(struct node *T){
   if (T == NULL)
     return 0;
   return (T->data + sum(T->left) + sum(T->right));
}
```

5. In a hash table with m slots using the chaining method, what is the expected number of items (keys) we need to map to the m slots until they store at least one item each. [Answer on the reverse side] [Hint: The crucial idea here is to define X_j equal to the number of items(keys) it takes to go from $(j-1)$ to j filled slots.] 4

Solution: Let X_j denote the number of items it takes to go from $(j-1)$ to j filled slots. X_j is a geometric random variable
If $\frac{j-1}{m}$ slots are filled, $\frac{m-j+1}{m}$ slots are empty. The probability of success for X_j is hence $\frac{m-j+1}{m}$.
Hence $E[X_j] = \frac{m}{m-j+1}$.

Let X denote the number of items (keys) we need to map to the m slots until they store at least one item each. Hence $X = \sum_{i=1}^{m} X_i$. Our goal is to find $E[X]$.

$$
\begin{aligned}
E[X] &= \sum_{i=1}^{m} E[X_i] \\
&= \frac{m}{m} + \frac{m}{m-1} + \cdots + \frac{m}{1} \\
&= m\left(1 + \frac{1}{2} + \cdots + \frac{1}{m}\right) \\
&= mH_m \\
&\leq m\left(1 + \log m\right) \\
&= O(m \log m)
\end{aligned}
$$