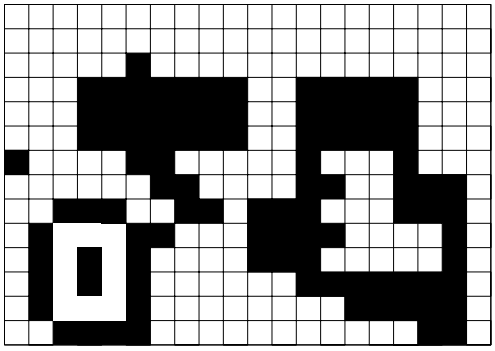
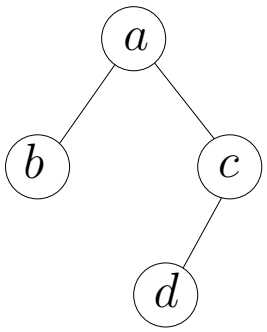
Assignment - 02
Advanced Data Structures and Algorithms
Indian Institute of Information Technology Kalyani

Due date: August 01, 2021

1. Write codes for insertion and deletion to a BST. Use your code to insert the following keys to an initially empty BST: 12, 5, 4, 7, 15, 20, 2, 24, and 5. Delete 15 and 20 from the BST now.
2. Modify your codes written for Prob-1 such that the search tree always grows as a height-balanced tree.
3. In the figure shown below, you have to count the number of connected components present. Consider the image as a 2D array, where te black squares are represented by 1 and the white squares by 0. Write a program, and your program should use both 4-N and 8-N connectivity separately for counting components.



4. In a representation of binary trees, a node **a** with left child and right child null is written as (a()()). The folowing tree is presented by the string: (a((b()()))((c((d()()))()))).
   Given a string S in that form, you have to check if S represents a BST or not. Write a code for the same.



DFS and BFS Traversal:

You may use these traversal methods to solve Prob 3. In 4-N, square a is adjacent to square b if b is immediate *south* or *north* or *east* or *west* square of a. In 8-N, we include *north-east*, *south-east*, *south-west* and *north-west* with the definition of 4-N.

Algorithm 1: Depth-First-Search

Input: G(V, E)

```
1 for all v ∈ V do
2     v_status ← WHITE
3 for each v ∈ V do
4     if v_status == WHITE then
5         DFS(v)
```

Procedure DFS(v)

```
1 v_status ← GRAY
2 for each u ∈ Adj(v) do
3     if u_status == WHITE then
4         DFS(u)
5 v_status ← BLACK
```

Algorithm 2: Breadth-First-Search

Input: G(V, E)

```
1 for all v ∈ V do
2     v_status ← WHITE
3 for each v ∈ V do
4     if v_status == WHITE then
5         BFS(v)
```

Procedure BFS(w)

```
1 w_status ← GRAY
2 ENQUEUE(w, Q)
3 while Q ≠ φ do
4     v ← DEQUEUE(Q)
5     for each u ∈ Adj(v) do
6         if u_status == WHITE then
7             u_status ← GRAY
8             ENQUEUE(u, Q)
9     v_status ← BLACK
```