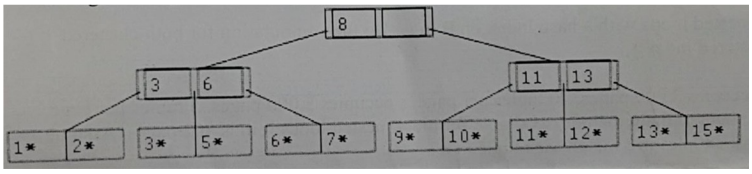
Data Structures

Practice Questions

Spring 2018

Problem 1. Consider the following intermediate state of a 2-3 tree



Perform the following operations on the tree
1) Lookup for value 8
2) Lookup for value 11
3) Insert elements 16,17,18
4) Insert value 0
5) Delete Elements 11,12,16 (Assume deletion algorithm tries to merge with the right sibl
if one exists first)

Problem 2. The incidence matrix of a directed graph G = (V,E) where V is the set of vertices and E is the set of edges of the graph with no self-loops is a $|V| \times |E|$ matrix **B** = b_{ij} such that
1) b_{ij} = -1 if edge j leaves at vertex i
2) b_{ij} = 1 if edge j enters vertex i
3) 0 otherwise
Describe what the entries of the matrix product BB^T represent, where B^T is the transpose of B .

Problem 3. Diameter for a tree is defined as longest path between any two nodes of a tre
Show that for any for any node of a tree the fartest node from it is one of the two ends of
diameter .

Problem 4. Consider the following DFS algorithm marking starting and end stamps for a node, *u.d* here represents start time for a node and *u.f* represents end time for a node .

```
DFS(G)
1  for each vertex u ∈ G.V
2      u.color = WHITE
3      u.π = NIL
4  time = 0
5  for each vertex u ∈ G.V
6      if u.color == WHITE
7          DFS-VISIT(G, u)

DFS-VISIT(G, u)
1  time = time + 1            // white vertex u has just been discovered
2  u.d = time
3  u.color = GRAY
4  for each v ∈ G.Adj[u]      // explore edge (u, v)
5      if v.color == WHITE
6          v.π = u
7          DFS-VISIT(G, v)
8  u.color = BLACK            // blacken u; it is finished
9  time = time + 1
10 u.f = time
```

All the time stamps (start time and end time) of node are written in increasing order . Start time stamps are then replaced with '('(open bracket) and end time stamp with ')' (close bracket), note that all time stamps will be unique .
Show that the sequence of brackets thus formed will be a balanced bracket expression .

Problem 5. A directed graph G = (V,E) is singly connected if for every vertex pair u,v there exists at most one simple path from u to v. Give and efficient algorithm to find whether or not a directed graph is singly

Problem 6. Give an efficient algorithm which for a given undirected graph G = (V,E) prints all nodes which are part of any cycle .

Problem 7. Suppose that a graph G has a minimum spanning tree already computed. How quickly can we update the minimum spanning tree if :
1) we add a new node and indent edges on G
2) modify weights of a few edges

Problem 8. Professor Lin claims that the algorithm for strongly connected components would be simpler if it used the original (instead of graph with reverse edges) graph in the second depth first search and scanned the vertices in order of increasing finishing times. Is this claim of professor correct ?