COL 351 : Analysis and Design of Algorithms

Tutorial Sheet - 4

Question 1 Complete the pseudo-code below to obtain an $O(|V| + |E|)$ time algorithm for computing cut-vertices of an n-vertex undirected (possibly unconnected) graph $G = (V, E)$.

```
VISITED[v] ← True;
HIGH-POINT[v] ← LEVEL[v];
foreach w ∈ N(v) do
    if (VISITED[w] = False) then
        Set PARENT[w] ← v  and  LEVEL[w] = 1 + LEVEL[v];
        ________________;
        ________________ ← ______________________________________;
        if ________________ ≥ ___________ then
            ___________________________;
        end
    else if (PARENT[w] ≠ v) then
        HIGH-POINT[v] ← min{LEVEL[w], HIGH-POINT[v]};
    end
end
```

Procedure DFS(v)

```
Let (v_1, ..., v_n) be any ordering of vertices of G;
for i = 1 to n do
    VISITED[v_i] ← False  and  IS-CUT-VERTEX[v_i] ← False ;
end
for i = 1 to n do
    if (VISITED[v_i] = False)  then  LEVEL[v_i] ← 0  and  Invoke DFS(v_i) ;
end
if (v_1 has two children)  then
    ____________________________;
end
```

Procedure Compute-Cut-vertices(G)

Question 2 A topological order of a DAG is a linear ordering of its vertices such that for every directed edge (x, y), x comes before y in the ordering. Let $G = (V, E)$ be a DAG on n vertices, and $L = (v_1, \ldots, v_n)$ be a list of vertices of G satisfying

$$\text{FINISH-TIME}(v_1) > \text{FINISH-TIME}(v_2) > \cdots > \text{FINISH-TIME}(v_n)$$

with respect to some DFS traversal of G. Then prove that L is a topological ordering of G.

Question 3 Let $G = (V, E)$ be a DAG and s be a source vertex such that all vertices of G are reachable from s. Let y and z be any two vertices in G satisfying that there is a path from y to z. Prove that the following two properties hold true for each DFS traversal carried out from s in G.

1 : FINISH-TIME(z) < FINISH-TIME(y).

2 : If START-TIME(y) < START-TIME(z), then z must be a descendant of y in the DFS tree.

Show that if G was a general directed graph, and vertices y, z satisfies that there is a path from y to z in G. Then, START-TIME(y) < START-TIME(z) does not imply that z is a descendant of y in the DFS tree.

Question 4 Prove that any n vertex undirected graph contains at most $n - 1$ bridge-edges and at most $n - 2$ cut-vertices.

Question 5 Let $A = (a_1, \ldots, a_n)$ and $B = (b_1, \ldots, b_m)$ be two sequences, where a_i's and b_j's are positive integers. Devise an $O(mn)$ time algorithm to compute a **longest common increasing subsequence** of A and B.

Question 6 Let $A = (a_1 \ldots a_n)$ and $B = (b_1 \ldots b_m)$ be two strings, where a_i's and b_j's are English alphabets. Devise an $O(mn)$ time algorithm to compute a **longest common substring** of A and B.