CS 375 HW3

Baptiste Saliba

April 7, 2019

"I have done this assignment completely on my own. I have not copied it, nor have I given my solution to anyone else. I understand that if I am involved in plagiarism or cheating I will have to sign an official form that I have cheated and that this form will be stored in my official university record. I also understand that I will receive a grade of 0 for the involved assignment for my first offense and that I will receive a grade of "F" for the course for any additional offense."

Question 1:

$I1 : 10\$/lb \quad I2 : 7\$/lb \quad I3 : 6\$/lb \quad I4 : 8\$/lb$

Total weight: 30

Order: I5 (25 pounds left), I4 (15 pounds left), I2 (0 pounds left)

$\$50 + \$80 + \$105 = \$235$

Question 2:

	Y_i	$P(1)$	$L(2)$	$A(3)$	$T(4)$	$E(5)$
X_i	0	0	0	0	0	0
$A(1)$	0	$0(<)$	$0(<)$	$1(=)$	$1(<)$	$1(<)$
$P(2)$	$\mathbf{0}$	$1(=)$	$1(<)$	$1(<)$	$1(<)$	$1(<)$
$P(3)$	0	$\mathbf{1(=)}$	$1(<)$	$1(<)$	$1(<)$	$1(<)$
$L(4)$	0	$1(\wedge)$	$\mathbf{2(=)}$	$\mathbf{2(<)}$	$\mathbf{2(<)}$	$2(<)$
$E(5)$	0	$1(\wedge)$	$2(\wedge)$	$2(<)$	$2(<)$	$\mathbf{3(=)}$

LCS: PLE
Path: P->L->L->L->E

Question 3:

```
DFS-Visit(G,U)
    time = time+1;
    u.d =time;
    u.color = Gray;
    for each v ∈ G.Adj[u]
        if v.color == White
            v.π = u;
            DFS-Visit(G,v);
        if v.color == Gray
            cycle = true
    v.color = black
    time = time+1
    u.f = time
```

Note: 'cycle' is a global variable. If you visit a node that's grey it means it's already been visited and you can return to the current node from there so it's a cycle

Question 4:

```
int count = 0
while(G.V ≠ ∅)
    S = G.V[0]
    BFS(G,S)
BFS(G,S)
    for each v ∈ G.V - {s}
        v.color = White
        v.d = ∞
        v.π = NIL
    s.color = Gray
    s.d = 0
    Q = ∅
    Enqueue(Q,S)
    //This is checking if there are adjacent node.
    //If there aren't then it's not a connected component
    //so don't increment count
    if(adj[s] ≠ ∅)
        count++
    while(Q ≠ ∅)
        u = Dequeue(Q)
        for each v in adj[u]
            if (v.color == White)
                v.color = Gray
                v.d = u.d+1
                v.π = u
                Enqueue(Q,V)
        u.color = black
        G.V = G.V - {u}
```

Question 5:

a) 1, 2, 3, 4, 5, 6, 7

b) 1, 2, 5, 6, 3, 7, 4

Question 6:

```
int SumDegree(G)
    int count = 0
    for each node v in G.V
        for each node a in v.adj
            count++
    return count
```

Time Complexity: $O(|V| + |E|)$

Reasoning: The two for loops are iterating through all the edges in the graph. If the number of edges ($2|E|$) is smaller than the number of vertices then the time complexity is dominated by the number of vertices. If the number of edges is larger than the number of vertices then the time complexity is dominated by the number of edges.