HW 1

Xinhao Luo

Tuesday 12th May, 2020

1 Problem 1

- Answer: 2^n
- Prove By Induction
  - Observe the program, we have: $T(n) = 1 + (T(0) + T(1) + T(2) + T(3) + ... + T(n-1))$

Base Case $T(0) = 1 = 2^0$

Induction Step Assume $T(n) = 2^n$, Prove $T(n+1) = 2^{n+1}$, Let:

*
$$\begin{aligned} S &= 2^0 + 2^1 + 2^2 + ... + 2^n \\ 2S &= 2^1 + 2^2 + 2^3 + ... + 2^{n+1} \\ 2S - S &= 2^1 - 2^1 + 2^2 - 2^2 + 2^3 - 2^3 + ... + 2^{n+1} - 2^0 \\ S &= 2^{n+1} - 1 \end{aligned} \tag{1}$$

* We have $T(n+1) = 1 + 2^{n+1} - 1 = 2^{n+1}$

- Proved $T(n) = 2^n$

2 Problem 2

- It is observed that $T(n) = n + (n-1) + (n-2) + (n-3) + ... + 0 = \frac{n^2+n}{2}$

Base Case $T(0) = 0$

Induction Step Assume $T(n) = \frac{n^2+n}{2}$, prove $T(n+1) = \frac{(n+1)^2+(n+1)}{2}$

$$\begin{aligned}
T(n+1) &= 1 + 2 + 3 + ... + (n+1) \\
&= \frac{n^2+n}{2} + n + 1 \\
&= \frac{n^2+n+2n+2}{2} \\
&= \frac{(n+1)^2+(n+1)}{2}
\end{aligned} \tag{2}$$

- Proved that $T(n) = (n^2+n)/2$

3 Problem 3

a) Prove by Induction

Base Case when $u = 0$, Sum of $d(u) = 0 = 2|E|$

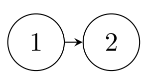
Induction Step Assume claim is true for any graph with u nodes, prove it is still applied for $u + 1$.
The number of degree for $u + 1$ is $2(n + 1) = 2n + 2$
Every time we have a new edge, it will connect two vertices, and each vertices will increase degree by 1. In total, an edge increase degree by 2, so that degrees will be equal to 2x edges

b) Prove by contradiction

- Assume there are odd number of vertices has odd degrees
- The rest of the nodes will be even degrees for sure.
- An odd number times an odd number will always be odd, and either even number of an odd number times and odd number, the result will be even
- The total degree will be an odd number plus a even number, which will be odd number
- the total degree of a graph will always a even number, which contradicted with the previous conclusion

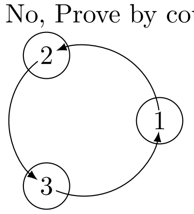
conclusion The original statement must be true: even number of vertices has odd degrees

c) a) No, Prove by counter example



The indegree is 1 here.

b) No, Prove by counter example



Each node has only one indegree, but the total number of vertices is odd as well

4 Problem 4

Algorithm

Step 1 For each node in graph G_1, use BFS to find the shortest path to node s, and save the distance into a Hashtable (with O(1) average access time). Use BFS and find shortest distance and store for graph G_2 repectively.

Step 2 For each edge E_1, we may find its distance D_1 to s, and distance D_2 to t in G_1. The total shortest distance of given edge D can be calculated as following:

$$D = D_1 + D_2 + 1 \tag{3}$$

We may keep replace D if current edge in E' is smaller than the previous one so that we will always have the smallest path

Step 3 We compared the shortest distance D we have from previous step, with the given number. **Return true if same, false otherwise**

5 Problem 5

Algorithm

Step 1 Start from a random node, we use BFS to iterate the graph

Step 2 Each time we have go thought all neighbours of one nodes, **we mark it with x if it is odd layer, v if it is even layer**.

Step 3 When marking the node, **a node has already marked with the same mark, return false**

Step 4 Check if all nodes has been checked. **If yes, return true**