GTU Department of Computer Engineering

CSE 222 / 505 – Spring 2022

Homework 5 Report

Ahmet USLUOGLU

1801042602

Problem 1 - Solutions

1-a) Calculate the total depth of nodes in a complete binary tree. * Assume tree is perfect

- Total depth of the tree is the sum of all depth's of all nodes.

Total Depth at height h is = height of level (h) × nodes at level (2^{h-1})

Sum of all depths in all heights = $\sum_{n=0}^{h} n \cdot 2^{n-1}$

$$F(h) = \sum_{n=0}^{h} n \cdot 2^{n-1} = 2^h \cdot (h-1) + 1$$

h → 1

F(h) → 1   5   17

1-b) Calculate the average number of comparisons for a successfull search operation in a complete binary search tree. * Assumed tree is perfect

- To calculate the average number of comparisons we must find the number of comparisons for each element and add them. Then divide the total number of comparisons to number of nodes.

Total number of comparisons is = $\sum_{i=1}^{n} h_i \times \frac{1}{n}$

Probability of each item being searched is $\frac{1}{n}$ where n is number of nodes.

h_i is the level of node i

For a complete binary tree number of total comparisons is equal to the total number of depths.

$$\sum_{i=1}^{n} h_i \times \frac{1}{n} = 2^h \cdot (h-1) + 1$$

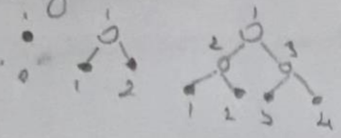
Total number of nodes $2^h - 1$.

Average number of comparisons = $\dfrac{\text{Total number of comparisons}}{\text{number of nodes}}$

$$F(n) = \frac{2^h \cdot (h-1) + 1}{2^h - 1} = \frac{h}{2^h - 1} + h - 1$$

1-c) Is there a restriction on the number of nodes in a full binary tree? What is the number of internal nodes and number of leaves in a n node binary tree?

There is always 1 more leaf node than the internal nodes in a Tree.



If tree has I internal nodes, number of leaves $L = I + 1$

Total number of nodes in a tree is sum of the internal nodes and leaves

Thus number of nodes in a tree of n nodes is $n = I + \underbrace{I+1}_{L}$

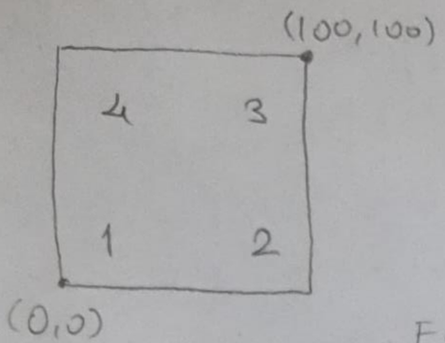
$n = 2I + 1$

Number of internal nodes is $I = \frac{n-1}{2}$

Number of leaves is $L = \frac{n+1}{2}$

There is no restriction for the number of nodes in a binary tree.

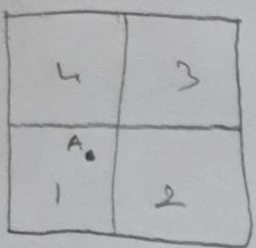
Problem 2 - Solution

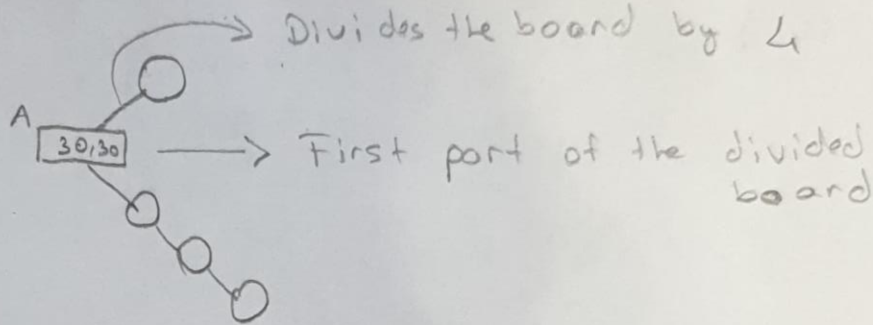


Notree
○ → Head node represents the 100x100 pixel board

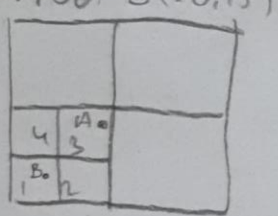
Each left branch diveds board at that index by 4.

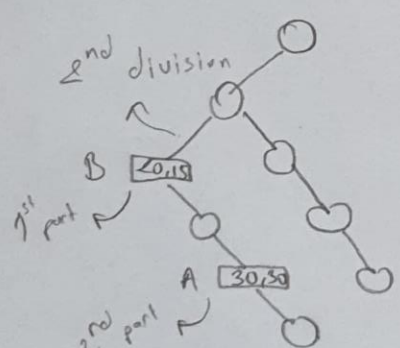
Add A(30,30)



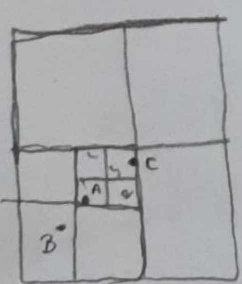


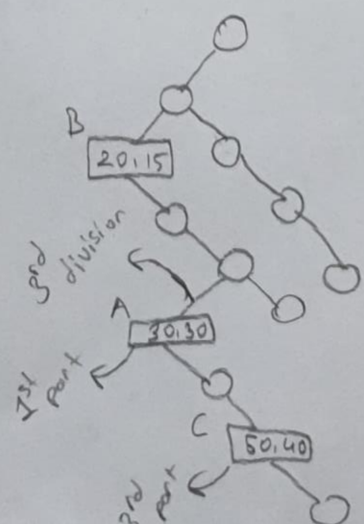
Add B(20,15)



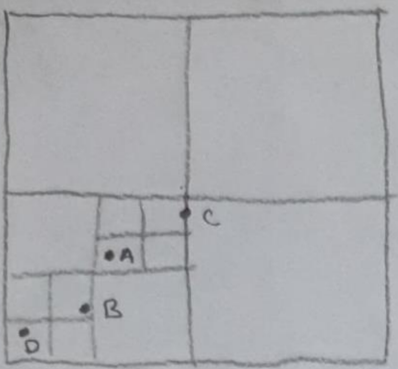


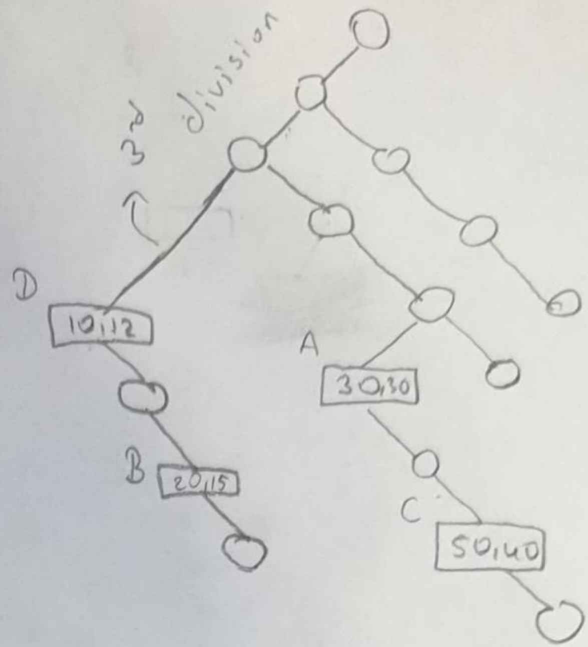
Add C(50,40)





Add D (10, 12)





Add E(40,20), F(25,60), G(15,25) → no more divisions will be made so adding all of them together.