Jaswal, Rajat: NUID-001443168

INFO 6205

PROGRAM STRUCTURES AND ALGORITHMS

FALL 2018

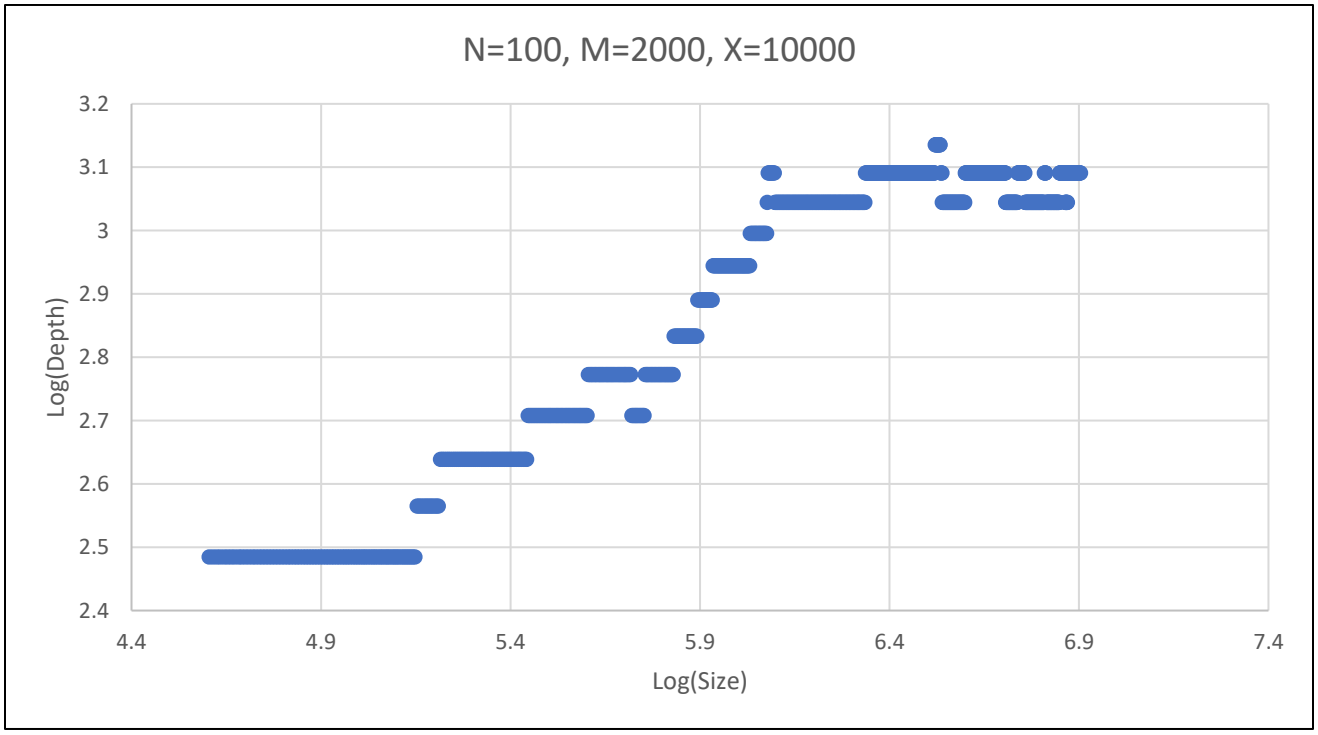
ASSIGNMENT 5

1. CONCLUSION:

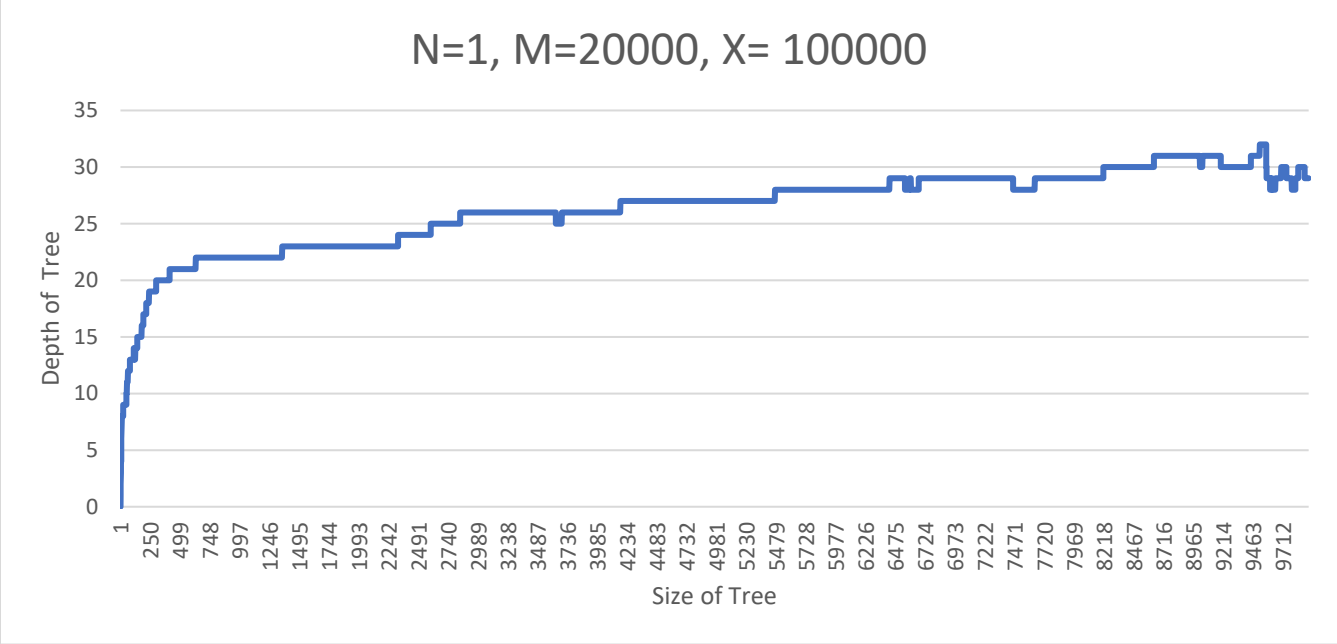
- The depth of a binary tree after X random insertion or deletions is directly proportional to the square root of N.
  i.e D=O(sqrt(N))
  Where N=size of tree.
  D= depth of the tree.
- If increasing the range of the node keys, the depth of binary insertion tree can tend to incline to log(N).
  ie. D=O(log(N).

2. GRAPHICAL REPRESENTATION

GRAPH-1



GRAPH-2



3. PROOF:

a) Let's consider the Graph 1 for analysis.
In the graph 1, I have plotted the graph among Log(D) and Log(N)
where D=depth of tree and N=size of tree.
However the graph follows a straight line, it maybe not visible clearly since some values are same for different N.
Hence we can assume a straight line
Log(D)=m*Log(N)+c.
where m=slope of line.
Let's calculate the slope of line from the graph.
Lets take coordinates 6.033 and 2.995
i.e almost 6 and 3.
Hence slope from graph=3/6.
Getting back to our equation
Log(D)=1/2*Log(N)+c;
Log(D)=Log(N^1/2)+c;

let's assume c as Log(c)
Log(D)=Log(N^1/2)+Log(c)
Log(D)=Log(c*N^1/2)
taking antilog on both sides.
D=c*N^1/2
Hence proved D is proportional to sqrt(N).

b) Let's now consider Graph 2.
Here I started from 1 and went upto 20000 and performed random 100000 insertion and deletions.
My graph and my values initially followed root(N) graph but later followed log(N) path.
I can prove that by my values.
I take 3 different values of D and N from my graph of larger N.
Let's consider D=k*log(N).
Hence for
1. N=9928, D=30.
   k1=D/log(N).
   k1=30/log(9928).
   k1=30/13.277=2.250.
2. N=3348, D=26.
   k2=26/11.7 = 2.200
3. N= 5600, D=28
   k3=28/12.45 = 2.245

We can clearly see k1~=k2~=k3
Hence the equation D=k*log(N) is true for larger N values.

HENCE PROVED

4. OBSERVATIONS

For higher values of N the depth is proportional to log(N).