CPSC 365 / ECON 365: Algorithms	Yale University
Discussion 4	
Out: February 24, 2022	Discussed: February 25, 2022

1 Comparing Algorithms

Suppose you are choosing between the following three algorithms:

(a) Algorithm A solves problems by dividing them into 5 subproblems of half the size, recursively solving each subproblem, and then combining the solution in linear time.

(b) Algorithm B solves problems of size n by recursively solving two subproblems of size $n-1$ and then combining the solutions in constant time.

(c) Algorithm C solves problems of size n by dividing them into nine subproblems of size $n/3$, recursively solving each subproblem, and then combining the solutions in $O(n^2)$ time.

2 Divide and Conquer

Consider an n-vertex complete binary tree T, where $n = 2^d - 1$ for some d.

- Each vertex v is labeled with a real number x_v, and all labels are distinct.
- You can only access a label x_v by *probing* the vertex v. E.g., you have access to a function PROBE such that PROBE$(v) = x_v$.
- A vertex v is a *local minimum* if x_v is smaller than x_w for all w adjacent to v.

Design an algorithm that returns a local minimum using only $O(\log n)$ probes.

Problems:

(a) Describe your algorithm in words. (You don't have to give a formal proof, but explain why your algorithm computes the correct answer.)

(b) Write a pseudocode for your algorithm to solve this problem.

(c) Let $T(n)$ be the number of probes made by your algorithm on a complete binary tree of n vertices. State the recurrence that $T(n)$ satisfies.

(d) Solve the recurrence. What is the number of probes made by your algorithm?