3.6 (12pts)

What is wrong with the following binary search algorithm?

function search(A, x, ℓ, r)
 comment find x in $A[\ell..r]$
 if $\ell = r$ **then return**(ℓ)
  else
   $m := \lfloor(\ell + r)/2\rfloor$
   if $x \leq A[m]$
    then return(search(A, x, ℓ, m))
    else return(search(A, x, m, r))

1. (16 points) Suppose $G = (V, E)$ is a directed acyclic graph represented by a adjacency lists. Divise a linear time algorithm that, given such a G, returns the length of the longest path in G. Prove your algorithm runs in $O(|V| + |E|)$-time

2. (16 points) Suppose $G = (V, E)$ is a directed graph represented by a adjacency lists. Divise a linear time algorithm that, given such a G, returns a list of all the source vertices of G. (Note, this list may be empty.) Prove your algorithm runs in $O(|V| + |E|)$-time. ***Hint:*** There is a simple solution that does not involve any DFS's or BFS's.

7.5 (20 pts) Hint: For (b) use a replacement argument.

Assume that you are given a set $\{x_1, \ldots, x_n\}$ of n points on the real line and wish to cover them with unit length closed intervals.

(a) Describe an efficient greedy algorithm that covers the points with as few intervals as possible.

(b) Prove that your algorithm works correctly.

7.6 (12 pts) Hint: Each of my answers uses S=4 and n=3.

Find counterexamples to the following algorithms for the knapsack problem. That is, find S, n, $s_1, \ldots, s_n$ such that when the rods are selected using the algorithm given, the knapsack is not completely full.

395. Put them in the knapsack in left to right order (the first-fit algorithm).

396. Put them in smallest first (the best-fit algorithm).

397. Put them in largest first (the worst-fit algorithm).