CS 180 Homework 1

February 4, 2015

1 Recurrences

(a) What is the next number in each of the following sequences?

    i. 1, 11, 21, 1211, 111221, 312211, **13112221**

    ii. 10, 9, 60, 90, 70, 66, **96**

    iii. 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, **58786**

    iv. 1, 2, 5, 15, 52, 203, 877, 4140, 21147, 115975, **678570**

(b) Give an explicit solution (nonrecursive expression) for $T(n)$, assuming that n is a power of d: $T(n) = kT(n/d) + f(n), T(1) = c$

$$T(n) = cn^{\log_d k} + \sum_{i=0}^{\log_d n - 1} k^i f(d^{\log_d n - i})$$

(c) For each of the fullowing recurrences, derive a solution $T(n) = \Theta(...)$, and determine which has the lowest asymptotic complexity.

    i. $T(n) = 8T(n/2) + 4n^2, T(1) = 1$

        $a = 8,\ b = 2,\ l = \log_b a = \log_2 8 = 4$
        $f(n) = O(n^2)$
        $T(n) = \Theta(n^4)$

    ii. $T(n) = 7T(n/2) + 18n^2, T(1) = 1$

        $a = 7,\ b = 2,\ l = \log_b a = \log_2 7 \approx 2.807$
        $f(n) = O(n^2)$
        $T(n) = \Theta(n^{\log_2 7})$

    iii. $T(n) = 143640 T(n/70) + O(n^2), T(1) = 1$

        $a = 143640,\ b = 70,\ l = \log_b a = \log_{70} 143640 = 2.795$
        $f(n) = O(n^2)$
        $T(n) = \Theta(n^{\log_{70} 143640})$

    The recurrence relation labeled iii. has the lowest time complexity.

(d) Give an explicit formula for $T(n) = \sum_{k=1}^{n-1} T(k)T(n-k), T(1) = 1$.

Initial values are as follows:

$T(1) = 1$
$T(2) = T(1) * T(1) = 1 * 1 = 1$
$T(3) = T(1) * T(2) + T(2) * T(1) = 1 * 1 + 1 * 1 = 2$
$T(4) = T(1) * T(3) + T(2) * T(2) + T(3) * T(1) = 1 * 2 + 1 * 1 + 2 * 1 = 5$
$T(5) = T(1) * T(4) + T(2) * T(3) + T(3) * T(2) + T(4) * T(1) = 14$
$T(6) = T(1)T(5) + T(2)T(4) + T(3)T(3) + T(4)T(2) + T(5)T(1) = 42$

These are the Catalan numbers which are given by $T(n) = \frac{2n!}{n!(n+1)!}$

2 iPhone Stress Testing (interview questions)

(a) If you have $k = 1$ iPhones, explain why you must use $\Omega(n)$ trials in the worst case.

If you have $k = 1$ iPhones, the only way to stress test the phones is by incrementing the floor by one until it breaks. This testing method always yields n trials, where n is the floor which causes iPhone to break.

(b) If you have $k = 2$ iPhones, develop a strategy that uses $O(\sqrt{n})$ trials to determine n.

The strategy here is to divide the iPhones in order to maximize each phone's testing capabilities. With 2 iPhones this can be done by taking the square root of the number of floors, which essentially gives a radix-like system for utilizing the iPhone: each phone will be used at most $\sqrt{n} - 1$ times. For example, with 100 floors, the most even division here is simply the decimal division, with the first one being used at most $\sqrt{n}$ times and the second one being used $\sqrt{n} - 1$. Therefore, the strategy's time complexity is $O(\sqrt{n})$.

(c) If you have $k \geq 3$ iPhones, develop a strategy that uses $O(kn^{1/k})$ trials.

A very similar strategy can be employed for any number of phones. In this case however, instead of splitting the trials evenly between 2 phones, the trials are split between k phones, which can be done in a very similar way. In this instance, the task can be split evenly by taking the kth root of the number of floors. As such each of the k phone will be used $\sqrt[k]{n}$ in the worst case. This strategy therefore has the time complexity $k\sqrt[k]{n}$, which can be written as $O(kn^{1/k})$.

3 Air Travel Networks

(a) Find the length of the route with the fewest hops from Dakar (DKR) to Los Angeles (LAX) using established flight connections. State the distance in hops.

The shortest route between the Dakar and Los Angeles airports is 2 hops.

(b) Finding a route with the most hops from DKR to LAX, visiting airports at most once, is an instance of the Longest Path problem. What is the time complexity of the Longest Path problem?

> The time complexity of the longest path problem in the general case is $O(d!2^d n)$ where d is the depth of the tree as a result of DFS and n is the number of vertices/nodes in the graph.

(c) The diameter $diam(G)$ of a graph G is the maximum of all pairwise distances (in hops) between nodes in any connected component; the average pairwise distance $apd(G)$ is the average of these distances. What are the diameter and apd of the airport graph? Question 3.8 in [KT] asks whether the ratio $diam(G)/apd(G)$ can be upper-bounded by a constant. Prove either there is such a bound, or that there is no such bound.

> By finding the maximum value in the distance matrix file, we can deduce that $diam(G) = 5$. Furthermore, by taking the half of the sum of all values in the distance matrix and dividing this by half of the number of airport distances, we can compute the average pairwise distance, giving us $apd(G) = 2.392$. The $diam(G)/apd(G)$ cannot be upper-bounded by a constant. If we assume that it could be bounded by a constant, this means that for any graph G there is minimum possible average pairwise distance relative to the diameter. However, given some graph G, it is always possible to add a node with that reduces the average pairwise distance. This is a contradiction, and as such the ratio cannot be upper-bounded by a constant.

(d) Determine whether this airport graph is scale-free by plotting the histogram for $P(d)$ and visually determining if $\log P(d)$ appears to be a linear function of d.

The histogram for $P(d)$ is as follows:
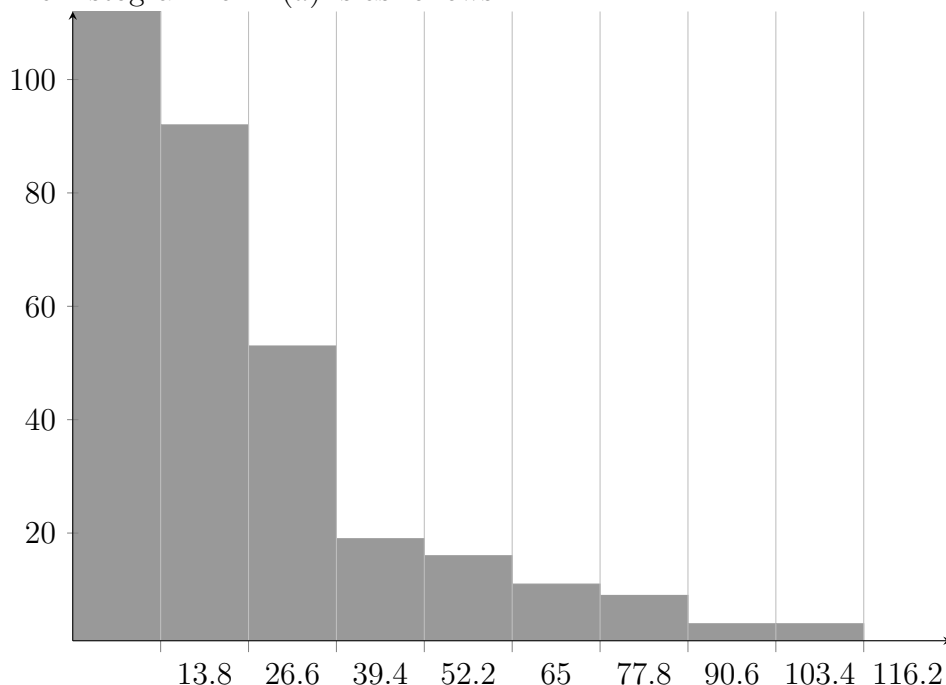
The graph can be considered a scale-free graph, as the $\log P(d)$ appears visually to be a linear function of d.

(e) A hub node is a node with high node degree, sometimes defined as being among the top 10% of nodes when they are ordered by their degree values. Identify the hub nodes of the graph, using this definition.

Some quick calculations in Excel reveal that the top 10% range of node degrees begins at 63.8. As such, the hub nodes are FRA, JFK, CDG, ATL, ORD, EWR, IST, DFW, DXB, IAH, FCO, LAX, YYZ, IAD, HKG, ICN, MUC, PEK, CLT, DEN, PHL, LAS, MSP, SIN, MAD, NIA, NRT, SFO, DOH, BOS, PVG, and SVO.

(f) Algorithm design (and interview questions) often requires order-of-magnitude estimates. Estimate how many flights there are every day between the 321 major airports. Using the 5th column of the airport connections.csv table, altogether the 4271 links between them are covered by airlines a total of 18334 times (i.e., each link is covered by approximately 4 airlines). However, some links are covered multiple times each day, some covered once every few days. Give an order-of-magnitude estimate for the number of flights each day.

As there is an average of 4 airlines covering each connection between airports, and an airline probably has about 3 flights on each link each day. We can estimate with 12 flights per link per day with about 4000 links, there are close to 48,000 flights each day.

(g) Assuming there are about 100 airplane disasters every year (for flights between major airports), based on the previous question, roughly what is the probability of a flight being a disaster? (Interview question)

With 48000 flights a day there are about 50,000 * 350 = 17,500,000 flights each year. With 100 airplane disasters each year disaster, this is a 1 in 175,000 chance of an airplane disaster each flight.

4   Articulation Points and Bridges

(a) Prove that s is an articulation point of G if and only if it has two or more children in T.

If a starting point s in the depth-first search tree T has two or more children, that means that the depth-first search had to back track all the way back to the root and down to another child. If there was another route to this child, it would not be necessary to return all the way back to the start. Furthermore, if a starting point s of a DFS tree in a graph had only one child, then its removal would simply be moving the root to the child, shortening the graph, but not disconnecting it.

(b) Prove that $v \neq s$ is an articulation point of G if and only if it has a child w such that there is no back edge from w or any descendant of w to a proper ancestor u of v.

Given that T forms a DFS tree of G, every edge is either a tree edge or a back edge. As such, when the vertex v is removed, the root of the tree T, s, and the vertex u are disconnected. If w is not a descendant of v, for every child u there is a back edge from a descendant of u to an ancestor of v, then after removing v for every w there is a path from the vertex to s, therefore v is not an articulationm point. It follows that v is an articulation point if and only if the property holds.

(c) Show how to compute all articulation points in $O(E)$ time

In order to compute all articulation points in $O(E)$ time, we must first run depth first search which is in $O(V + E)$ time, but as $E > V$, this reduces to $O(E)$ time. Once we have the DFS tree, we can test each individual point in the tree using the methods in parts (a) and (b) which take $O(1)$ and $O(V)$ time respectively. Thus the total complexity is $O(1 + V + E)$ which reduces to simply $O(E)$.

(d) Prove that an edge $e \in E$ is a bridge if and only if it does not lie on any simple cycle of G.

If we assume that edge e is in a simple cycle, then its removal does not affect the existence of any paths between two vertices in the simple cycle, as every potential path in a simple cycle can be written without the use of a particular edge. As such, if an edge is on a simple cycle then it is not a bridge. Further, if we assume that the edge e is not on a simple cycle and is not a bridge. If the edge is removed, there is still a path connecting the two points as e is not itself a bridge, this implies that the edges form a simple cycle, which is a contradiction. Therefore, if edge e is not in a simple cycle then it is a bridge.

(e) Find all articulation points and bridges in the airport graph. Which articulation points are hub nodes?

The articulation points in the airport graph are CAI, SEA, LIM, and SHJ. None of these articulation points are hub nodes.