CS 120: Intro to Algorithms and their Limitations

Problem Set 0

Due: September 13, 2023 **Denny Cao**

---

§1 Binary Trees

Answer 1.a (Recursive Programming)**.** In `ps0.py`. The algorithm is $O(n)$.

Proof. In each recursive call, the algorithm visits one node in the tree and adds the left and right subtrees to the stack, performing basic operations for each node in the stack. For a tree with n leaves, there will be n basic operations, and therefore the algorithms is $O(n)$. $\square$

Answer 1.b (Proof Warmup)**.**

Proof. Let the subtree that is produced when v is removed that contains v^*`.parent` be P, the subtree of v^*`.left` be L, the subtree of v^*`.right` be R. Let $|V(X)|$ denote the amount of vertices in a tree X.

v^* has at most 3 possible edges: $(v^*.\texttt{parent}, v^*), (v^*.\texttt{left}, v^*)$, and $(v^*.\texttt{right}, v^*)$. As v^* is a neighbor of v, one of these edges connects v and v^*. Thus, if v is removed, then one of these edges is removed, thereby creating at most 3 subtrees—one will contain v^* and will be the largest of the subtrees, and the others will be rooted at the neighbors of v other than v^*.

Thus, the number of vertices in largest subtree created by removing v will be the sum of subtrees that contain a neighbor of v^* (an edge between v^*) when v is removed:

- If v^*`.parent == v`, then $\phi(v) = |V(L)| + |V(R)| + 1$, as, if the subtree containing v^* is the largest subtree when v is removed, then the subtree will contain the subtrees rooted at v^*`.left` and v^*`.right`, as well as v^* itself.
- Otherwise, if v`.parent ==` v^*, then $\phi(v) = |V(P)| + |V(L)| + 1$ if v^*`.right ==` v or $\phi = |V(P)| + |V(R)| + 1$ if v^*`.left ==` v.

Thus, the subtree without v when v^* is removed that contains the most vertices (P, L, or R) is also included in the sum of $\phi(v)$. As $\phi(v)$ also counts v^*, $\phi(v)$ will always be greater than the greatest number of vertices amongst subtrees without v when v^* is removed. $\square$

Answer 1.c (Proofs by Contradiction)**.**

Proof. Assume for purposes of contradiction that, there exists a tree T of size n such that there exists a vertex v such that removing v from T results in at least one disjoint tree with size greater than $n/2$.

Let this subtree be denoted T', with size $\phi(v)$. Thus, $\phi(v) > n/2$. Let T' be the smallest possible value such that this is true; that $\phi(v) = \lfloor n/2 + 1 \rfloor$. As $\lfloor n/2 + 1 \rfloor \leq n/2 + 1$, it follows that $\phi(v) \leq n/2 + 1$.

From Question 1.b, we know that, if the neighbor of v in T'—v^*—is removed instead, then the largest subtree that does not include v will be less than $\phi(v)$. Let the size of this subtree be denoted $\phi(v^*)$. As $\phi(v^*) < \phi(v)$, it follows that $\phi(v^*) < n/2 + 1$, which means $\phi(v^*) \leq n/2$. As $\phi(v^*)$ is defined as the size of the largest subtree that does not include v when v^* is removed, it means that all other subtrees that do not include v when v^* is removed will also be at most $n/2$.

Let the subtree that includes v when v^* is removed be S. The total amount of vertices for T, $|V(T)|$, is the sum of $|V(T')| + |V(S)|$ as when v^* and v are reconnected, the tree T is produced. As $|V(T)| = n$ and $|V(T')| = \phi(v) = \lfloor n/2 + 1 \rfloor$, $n - \lfloor n/2 + 1 \rfloor = |V(S)|$:

$$n - \lfloor n/2 + 1 \rfloor = |V(S)|$$
$$n - 1 - \lfloor n/2 \rfloor = |V(S)|$$

Let $\lfloor n/2 \rfloor = c, c \in \mathbb{Z}$. Then, $n/2 = c \vee n/2 = c - 1/2$. If $n/2 = c$, then $|V(S)| = n - 1 - c = n/2 - 1$. If $n/2 = c - 1/2$, then $|V(S)| = n - 1 - c + 1/2 = n/2 - 1/2$. In both cases, $|V(S)|$ is less than $n/2$. ※

We reach a contradiction, as, from our assumption, if all subtrees not including v when v^* is removed have at most $n/2$ vertices, then the subtree with v has to have more than $n/2$ vertices. Thus, we have shown that for every tree T of size n, there exists a vertex v such that removing v from T results in disjoint trees that all have size at most $n/2$. □

Answer 1.d.

§2 Matchings and Induction

Answer 2.a.

Proof. Let the donors be $d_1, \ldots, d_n$ and the patients be $p_1, \ldots, p_n$, where $p_x = \text{incomp}(d_x)$ and $d_x = \text{incomp}(p_x)$. Without loss of generality, let d_i be paired with a patient it is compatible with, p_j, $j \neq i$. There are $n - 1$ possibilities for p_j as there are n patients and d_i is incompatible with p_i. There are now two cases for what d_j is matched to:

i) $(d_j, p_t), t \neq i$; d_j is matched with a patient that d_i is compatible with.

ii) (d_j, p_i); d_j is matched with the patient that d_i is incompatible with.

In the second case, there will be $f(n-2)$ ways to match the remaining $n - 2$ donors with patients, as each donor will have a patient they are incompatible with. Thus, in the second case, $f(n) = (n - 1)(1)(f(n - 2)) = (n - 1)f(n - 2)$.

As $(n-1)f(n-2)$ only accounts for the second case and $f(n)$ will be the sum of the number of possibilities in the first case as well as the second case, $f(n) > (n - 1)f(n - 2)$. □

Answer 2.b.

Proof. We consider the first case from the proof in Answer 2.a. There are $n-1$ matchings for d_i to a patient other than p_i, p_j. As case 1 considers when d_j is matched to a patient other than p_i and d_j is compatible with all remaining patients (p_j is already matched), there are $n - 1$ matchings for d_j.

Let d_j be matched to p_k. Then, there are now two cases for what d_k can be matched to:

i) (d_k, p_t),

$\square$

Answer 2.c.

Proof. By strong induction. Let $P(n)$ be the statement that $\frac{n!}{3} \leq f(n) \leq \frac{n!}{2}$, $n \geq 2$.

Base Case: $n = 2$. Since $\frac{2!}{3} = \frac{2}{3}$ and $\frac{2!}{2} = 1$, $P(2)$, or $\frac{2}{3} \leq f(2) \leq 1$, is true as $f(2) = 1$.

Inductive Step: Assume $P(k)$ for $2 \leq k < n$, $k, n \in \mathbb{N}$.

Then, by the inductive hypothesis:

$$\frac{(n-1)!}{3} \leq f(n-1) \leq \frac{(n-1)!}{2}$$
$$\frac{(n-2)!}{3} \leq f(n-2) \leq \frac{(n-2)!}{2}$$

Thus:

$$\frac{(n-1)!}{3} + \frac{(n-2)!}{3} \leq f(n-1) + f(n-2) \leq \frac{(n-1)!}{2} + \frac{(n-2)!}{2}$$

Multiplying by $(n-1)$, we obtain:

$$(n-1)\left(\frac{(n-1)!}{3} + \frac{(n-2)!}{3}\right) \leq (n-1)(f(n-1)+f(n-2)) \leq (n-1)\left(\frac{(n-1)!}{2} + \frac{(n-2)!}{2}\right)$$

From Answer 2.b:

$$\begin{aligned}
(n-1)\left(\frac{(n-1)!}{3} + \frac{(n-2)!}{3}\right) &\leq f(n) \leq (n-1)\left(\frac{(n-1)!}{2} + \frac{(n-2)!}{2}\right) \\
\frac{(n-1)(n-1)! + (n-1)(n-2)!}{3} &\leq f(n) \leq \frac{(n-1)(n-1)! + (n-1)(n-2)!}{2} \\
\frac{(n-1)(n-1)! + (n-1)!}{3} &\leq f(n) \leq \frac{(n-1)(n-1)! + (n-1)!}{2} \\
\frac{(n-1)!(n-1+1)}{3} &\leq f(n) \leq \frac{(n-1)!(n-1+1)}{2} \\
\frac{(n-1)!(n)}{3} &\leq f(n) \leq \frac{(n-1)!(n)}{2} \\
\frac{n!}{3} &\leq f(n) \leq \frac{n!}{2}
\end{aligned}$$

Thus, we have proved for n. Therefore, by strong induction, we have shown that $P(n)$ is true, $n \geq 2$. $\square$