CS3000: Algorithms & Data — Summer I '21 — Drew van der Poel

Homework 1
Due Friday, May 21 at 11:59pm via Gradescope

Name: Gabriel Peter
Collaborators:

- Make sure to put your name on the first page. If you are using the LaTeX template we provided, then you can make sure it appears by filling in the `yourname` command.
- This assignment is due Friday, May 21 at 11:59pm via Gradescope. No late assignments will be accepted. Make sure to submit something before the deadline.
- Solutions must be typeset in LaTeX. If you need to draw any diagrams, you may draw them by hand as long as they are embedded in the PDF. I recommend using the source file for this assignment to get started.
- I encourage you to work with your classmates on the homework problems. *If you do collaborate, you must write all solutions by yourself, in your own words.* Do not submit anything you cannot explain. Please list all your collaborators in your solution for each problem by filling in the `yourcollaborators` command.
- Finding solutions to homework problems on the web, or by asking students not enrolled in the class is strictly forbidden.

Problem 1. *Proof by Induction (8 points)*

Prove the following statement by induction: For every $n \in \mathbb{N}$, $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$

Solution:

Base: $n = 0$ such that $H(0) = 0^2 = \frac{(0)(0+1)(2(0)+1)}{6} = 0$ Pass!

Weak Induction:

$IH : H(k-1) = \sum_{i=1}^{n-1} i^2 = \frac{(n-1)(n+1-1)(2(n-1)+1)}{6} = \frac{(n)(n-1)(2n-1)}{6} = \frac{2n^3-3n^2+n}{6}$
$H(k) = \sum_{i=1}^{n} i^2 = \sum_{i=1}^{n-1} i^2 + n^2 = \frac{n(n-1)(2n-1)}{6} + n^2 = H(k-1) + n^2 = \frac{2n^3+3n^2+n}{6} = \frac{n(n+1)(2n+1)}{6}$ Pass!

Problem 2. *Mystery Code (11 points)*

You encounter the following mysterious piece of code.

Algorithm 1: Mystery Function

```
Function F(a, n):
    If n = 0 :
        Return (1, a)
    Else
        b = 1
        For i from 1 to 2n
            b = b · a
        (u, v) ← F(a, n − 1)
        Return (u · b/a, v · b · a)
```

(a) **[3 points]** What are the results of $F(a,3)$, $F(a,4)$, and $F(a,5)$. You do not need to justify your answers.

Solution:

$F(a,3) = (a^9, a^{16})$
$F(a,4) = (a^{16}, a^{25})$
$F(a,5) = (a^{25}, a^{36})$
$F(a,n) = (a^{n^2}, a^{(n+1)^2})$

(b) **[8 points]** What does the code do in general, when given input integer $n \geq 0$? Prove your assertion by induction on n.

Solution:

Claim: $F(a,n) = (a^{n^2}, a^{(n+1)^2})$ where $n \geq 0$
Base: $n = 0 \rightarrow F(a,0) = (1,a)$ AND $(a^{0^2}, a^{(0+1)^2}) = (1,a)$
Induction: We assume that $F(a,n-1) = (a^{(n-1)^2}, a^{(n)^2})$ is true for all $n \geq 0$

Provided from the pseudocode, we hit the else of our function and return:

Thus, $F(a,n) = (u * b/a, v * ba)$ where $b = a^{2n}$ and $(u,v) = F(a,n-1) = (a^{(n-1)^2}, a^{(n)^2})$
$F(a,n) = (u * b/a, v * ba) = (a^{(n-1)^2} * a^{2n}/a, a^{(n)^2} * a^{2n} * a)$
$= (a^{n^2-2n+1} * a^{2n} * a^{-1}, a^{n^2} * a^{2n} * a^1)$
$= (a^{n^2}, a^{(n+1)^2})$ via solving quadratic and is now equal to our claim! Fin.

Problem 3. *Stable Matching (14 points)*

(a) **[6 points]** State the matching you obtain from running the Gale-Shapley algorithm on the following instance:

hospital	1	2	3
h_1	d_2	d_1	d_3
h_2	d_2	d_3	d_1
h_3	d_1	d_3	d_2

doctor	1	2	3
d_1	h_2	h_3	h_1
d_2	h_3	h_1	h_2
d_3	h_1	h_3	h_2

Is the stable matching you found the only stable matching? If not, provide an example of another stable matching.

Solution:

$M = (h_1, d_2), (h_2, d_3), (h_3, d_1)$

(b) **[8 points]** Given a set of preferences for n doctors and n hospitals, consider the stable matchings found via the following processes:

(a) Run the standard Gale-Shapley algorithm with hospitals making offers to doctors. Let this matching be M_1.

(b) Run Gale-Shapley again, but this time flip the roles of the hospitals and doctors in the algorithm, so that the doctors make offers to the hospitals. Let this matching be M_2.

Prove the following claim:

If there is more than one stable matching, then $M_1 \neq M_2$.

To do this, you may use the following terminology and Lemma 1.7 from the text. Hospital h is a *valid partner* of doctor d if there is a stable matching the contains the pair (h, d) (and vice versa). Doctor d is the *best/worst valid partner* of h if every other valid partner is ranked lower/higher than d in h's preferences. When hospitals (doctors) propose in Gale-Shapley, each hospital (doctor) is paired with their best valid partner (Lemma 1.7).

Solution:

Proof by contradiction: Assume $M_1 = M_2$.

According to Lemma 1.7, "each hospital is paired with their best valid partner."

If $M_1 = M_2$ then the best matchings for hospitals is equivalent to the best matchings for doctors.
The term "best" asserts that it was the only matching possible.
Thus, we can assume they reduce to a single matching due to their strict equivalence. This is a contradiction as there are now not two stable matching's but one that occurs twice.

Therefore: $M_1 \neq M_2$, if there is more than one stable matching.

Problem 4. *Asymptotic Order of Growth (18 points)*

(a) **[10 points]** Rank the following functions in increasing order of asymptotic growth rate. That is, find an ordering $f_1, f_2, \ldots, f_{10}$ of the functions so that $f_i = O(f_{i+1})$. No justification is required.

$$\begin{array}{ccccc} n^3 & \sqrt{n} & n! & 12^n & \log_2(n!) \\ 2^{4n} & 100n^{3/2} & 10n & 2^{\log_3 n} & \log_2^3 n \end{array}$$

Solution:

$\sqrt{n}, 2^{\log_3 n}, 10n, \log_2^3 n, 100n^{3/2}, \log_2(n!), n^3, 12^n, 2^{4n}, n!$

(b) **[8 points]** Suppose $f(n), g(n), h(n)$ are non-decreasing, non-negative functions. Decide whether you think the following statement is true or false and give a proof or a counterexample.

If $f(n) = \Omega(h(n))$ and $g(n) = O(h(n))$, then $f(n) = \Omega(g(n))$.

Solution:
Given that $f(n) = \Omega(g(n)) \rightarrow f(n) \geq c * g(n)$ and $f(n) = O(g(n)) \rightarrow f(n) \leq c * g(n)$ from Lecture 3/4 notes.

$f(n) = \Omega(h(n)) \rightarrow f(n) \geq c * h(n)$
$g(n) = O(h(n)) \rightarrow g(n) \leq c * h(n)$
Therefore, if we merge: $f(n) \geq c * h(n) \geq g(n)$

Transitively, $f(n) \geq g(n) \rightarrow f(n) = \Omega(g(n))$