Problem Set 1 (due Friday, September 21, 5:59 PM)

Instructions:

- The assignment is due at the time and date specified. Late assignments will be accepted, up until Saturday, September 22, 5:59 PM. *Note, however, that you can use at most 3 late days for your problem set and programming assignment submissions throughout the course of the term.*
- We encourage you to attempt and work out all of the problems on your own. You are permitted to study with friends and discuss the problems; however, *you must write up you own solutions, in your own words.*
- Please refrain from searching online or asking your peers or other students for solutions. The best way to learn the material is to attempt the problem yourself, and if you are stuck, identify where and why you are stuck and seek help to overcome the associated hurdles.
- If you do collaborate with any of the other students on any problem, please *list all your collaborators in your submission for each problem.*
- We require that all homework submissions be neat, organized, and *typeset.* You may use plain text or a word processor like Microsoft Word or LaTeX for your submissions. If you need to draw any diagrams, however, you may draw them with your hand.

1. (4 points) Instability of a local improvement algorithm

As we discussed in class, there may be several approaches to finding a stable matching. One reasonable approach is the following. Suppose we start with an arbitrary matching, and then repeat the following step until there are no unstable pairs.

- If there exists unstable pairs (m, w) and (m', w') such that m prefers w' over w and w' prefers m over m', then replace the pairs by (m, w') and (m', w).

The above local improvement algorithm is fairly natural. But it does not work! Cycles may occur, especially if you choose the "wrong" unstable pairs to swap, causing the algorithm to loop forever. Consider the following preference lists with 3 women A B, C, and 3 men U, V, and W.

A	B	C	U	V	W
U	W	U	B	A	A
W	U	V	A	B	B
V	V	W	C	C	C

For the above preference list, show that there exists a 4-step cycle in the local improvement algorithm, starting with the matching $\{(A, U), (B, V), (C, W)\}$.

2. (4 points) Stable Matching with ties in preference lists

The Stable Matching Problem, as we discussed in class, assumes that all resource providers and resource seekers (say, hospitals and students) have a fully ordered list of preferences. Here, we will consider a version of the problem in which one of the parties, say the students, can have equal preference for certain options.

As before we have a set H of n hospitals, and a set S of n students. Assume each hospital and each student ranks the members of the other set. The hospitals have a strict preference ordering of the students; given students s_1 and s_2, hospital h either prefers s_1 to s_2, or vice versa. On the student side, we allow for ties. A student s may prefer hospital h_i over hospital h_j, have equal preference for h_j and h_k, and prefer h_j and h_k over h_l.

With equal preferences allowed in the ranking of hospitals by students, we define the notion of instability as follows:

An *instability* in a perfect matching S consists of a hospital h and a student s, such that each of h and s has a *strictly higher* preference for each other than their partner in S.

In this variation of the problem, we ask the following.

True or **False**: Does there always exist a perfect matching with no instability?

If your answer is true, then give an algorithm that is guaranteed to find a perfect matching with no instability, and prove the correctness of your algorithm.

If your answer is false, then give an instance of a set of hospitals and students with preference lists for which you argue that every perfect matching has an instability.

3. (4 points) Fibonacci numbers

The *Fibonacci numbers* are defined by the following recurrence:

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1, \text{ and} \\ F_i &= F_{i-1} + F_{i-2} \text{ for } i \geq 2. \end{aligned}$$

Prove by induction that for any $n \geq 2$, F_n equals $A_{1,1}^{n-1}$, where A^n is the nth power of the following matrix:

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

4. (4 points) Ordering functions

Arrange the following functions in order from the slowest growing function to the fastest growing function. Briefly justify your answers. (*Hint:* It may help to plot the functions and obtain an estimate of their relative growth rates. In some cases, it may also help to express the functions as a power of 2 and then compare.)

$$\sqrt{n} \quad n\sqrt{\lg n} \quad 2^{\sqrt{\lg n}} \quad (\lg n)^2$$

5. (2 × 2 = 4 points) Properties of asymptotic notation

Let $f(n)$, $g(n)$, and $h(n)$ be asymptotically positive and monotonically increasing functions. For each of the following statements, decide whether you think it is true or false and give a proof or a counterexample.

(a) $f(n) + g(n) = \Theta(\max\{f(n), g(n)\})$.

(b) If $f(n) = O(g(n))$, then $2^{f(n)}$ is $O(2^{g(n)})$.