CS / MCS 401 – Computer Algorithms I
Spring 2025
Problem Set 1

Lev Reyzin

Due: 2/5/25 by the beginning of class

1. Arrange the following functions in ascending order of asymptotic growth rate; that is, if function $g(n)$ immediately follows function $f(n)$ in your list, then it should be the case that $f(n)$ is $O(g(n))$: $2^{\sqrt{\log n}}, 2^n, n^{4/3}, n(\log n)^3, n^{\log n}, 2^{2^n}, 2^{n^2}$. (There is no need to justify your answer.)

2. Consider a stable marriage instance with men $M = \{m_1, m_2, m_3\}$ and women $W = \{w_1, w_2, w_3\}$ having the following preference lists:

$m_1 : w_1 > w_2 > w_3$	$w_1 : m_2 > m_1 > m_3$
$m_2 : w_2 > w_1 > w_3$	$w_2 : m_1 > m_2 > m_3$
$m_3 : w_1 > w_2 > w_3$	$w_3 : m_1 > m_2 > m_3$

a. If these preferences are given to the Gale-Shapley algorithm, what is the resulting set of marriages?

b. Now, suppose w_1 decides to lie to the algorithm about her preferences and instead submits the list $w_1 : m_2 > m_3 > m_1$ to the Gale-Shapley algorithm (with everyone else's preference lists remaining unchanged). What is the resulting set of marriages? Did w_1 get a better, worse, or same partner according to her true preference by lying this way compared to the result in part a.?

3. Consider an instance of the stable marriage problem for $n = 4$, with the men as $M = \{m_1, m_2, m_3, m_4\}$ and the women as $W = \{w_1, w_2, w_3, w_4\}$. Consider the following preferences (each list is ranked from first to last choice):

men's preferences	women's preferences
$m_1 : w_1 > w_2 > w_3 > w_4$	$w_1 : m_2 > m_3 > m_4 > m_1$
$m_2 : w_2 > w_3 > w_1 > w_4$	$w_2 : m_3 > m_4 > m_1 > m_2$
$m_3 : w_3 > w_1 > w_2 > w_4$	$w_3 : m_4 > m_1 > m_2 > m_3$
$m_4 : w_1 > w_2 > w_3 > w_4$	$w_4 : m_1 > m_2 > m_3 > m_4$

What is the matching produced by the Gale-Shapley algorithm? For each man/woman, also designate their partner's *ranking* on his/her preference list, respectively.

4. Is it possible for the Gale-Shapley algorithm to assign two different men to their respective last-choice woman? Prove your answer correct.

5. Consider extending the stable marriage problem to the case of fellowship programs in a given specialty (which doctors can complete as additional training after their residency). Each of m doctors ranks all the fellowship programs. Each of n programs ranks its applicants. Each doctor can be admitted to at most one fellowship program, but each program has a different number of slots to fill in its class. Assume that there may be more applicants than total slots, so some doctors might not be assigned to any fellowship.
An assignment of doctors to programs is stable if neither of the following two instabilities exist:

- There are doctors d and d', and a program p, so that d is assigned to p, and d' is assigned to no program, and p prefers d' to d.
- There are doctors d and d', and programs p and p', so that d is assigned to p, and d' is assigned to p', and p prefers d' to d, and d' prefers p to p'.

Prove that a stable assignment of doctors to fellowship programs always exists.

6. Consider the following problem. You are given array X consisting of integers $X[1], X[2], \ldots X[n]$, and you need to output an $n \times n$ matrix Y in which the entries $Y[i,j] = X[i] + X[i+1] + \ldots + X[j]$ for $i < j$ (for $i \geq j$, $Y[i,j]$ may contain any value). Consider the following algorithm for this problem.

Algorithm 1 Compute Y (given input X, an array of n integers)

```
initialize Y to an n × n matrix of 0s
for i = 1 to n do
    for j = i + 1 to n do
        for k = i to j do
            Y[i, j] = Y[i, j] + X[k]
        end for
    end for
end for
return Y
```

i. Give a function $f(n)$ that bounds the worst-case running time of Algorithm 1 as $\Theta(f(n))$. Note: to do this, you must show that the worst-case running time of Algorithm 1 is both $O(f(n))$ and also $\Omega(f(n))$.

ii. Design an algorithm for this problem with asymptotically faster running time than that of Algorithm 1. Explain your answer.