Homework 1 (100 Points)

Practicing Topics from Chapters 1, 2, 3

Posted on Fri 10/7, due on Sun 10/16 by 11:00pm (on canvas) - **No Late Submission**

1. *(25 Points)* **Chapter 1 - Stable Marriage.** Consider the following preference lists for men and women:

Man #	Preference List
1	1 2 3 4
2	2 1 4 3
3	3 4 1 2
4	4 3 2 1

Woman #	Preference List
1	4 3 2 1
2	3 4 1 2
3	2 1 4 3
4	1 2 3 4

   (a) *(5 Points)* Find the man optimal stable matching. Show some intermediate steps of your answer.

   (b) *(5 Points)* Find the woman optimal stable matching. Justify your answer

   (c) *(5 Points)* Suppose that in another problem setting all men have identical preferences. How many steps does it take for the algorithm to converge? Justify your answer.

   (d) *(10 Points)* **Truthfulness in Stable Marriage.**

   The classic stable matching problem assumes that all men and women state their true preferences. Let us now consider a version of the problem in which people can lie about their preference. More specifically, let us consider the following restricted scenario.

   Consider three men m_1, m_2, m_3 and three women w_1, w_2, w_3. Everybody states their true preferences except for woman w_3, who can lie to some extent: she can look at her true preference list and switch the place of two men in her list with each other (she can only flip a pair). Woman w_3 has no control over the true preference lists (her own or other people's). The only decisions she can make are: (1) whether to lie or not (in the way mentioned above, *i.e.,* switching a pair of men in her list); and (2) if she decides to lie, she can also choose which pair (among all three pairs of men) to switch in her list. The GS algorithm then runs with input the declared preference lists and produces as output a stable matching.

   The question is: can woman w_3 get a better match by lying about her preferences? Resolve this question by doing one of the following:

      i. *(5 Points)* Prove that, for any set of preference lists, switching the order of a pair on the list cannot improve a woman's partner in the GS algorithm; or

      ii. *(5 Points)* Give an example of a set of preference lists for which, there is a switch that would improve the partner of a woman who switched preferences. And if she decides to lie, which pair of men should she switch in her preference list?

2. *(40 Points)* **Chapter 2 - Running Time Analysis.**

(a) *(20 Points)* **Growth Rates** Order the functions in ascending order of 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 $\mathcal{O}(g(n))$.

$$g_1(n) = 2^{\sqrt{logn}}, g_2(n) = 2^n, g_3(n) = n(logn)^3, g_4(n) = n^{\frac{4}{3}}, g_5(n) = n^{logn}, g_6(n) = 2^{2^n}, g_7(n) = 2^{n^2}$$

(b) *(20 Points)* Consider the problem below:

You're given an array A consisting of n integers $A[1]$, $A[2]$, ..., $A[n]$. You'd like to output a two-dimensional n-by-n array B in which $B[i,j]$ (for $i < j$) contains the sum of array entries $A[i]$ through $A[j]$–that is, the sum $A[i] + A[i+1] + ... + A[j]$. (The value of array entry $B[i,j]$ is left unspecified whenever $i \geq j$, so it doesn't matter what is output for these values.)

Here's a simple algorithm to solve this problem:

```
for i= 1, 2,..., n do
    for j= i+1, i+2,..., n do
        Add up array entries A[i] through A[j]
        Store the result in B[i, j]
    end for
end for
```

(1) For some function f that you should choose, give a bound of the form $\mathcal{O}(f(n))$ on the running time of this algorithm on an input of size n (i.e. a bound on the number of operations performed by the algorithm).

(2) For this same function f, show that the running time of the algorithm on an input of size n is also $\Omega(f(n))$. (This shows an asymptotically tight bound of $\Theta(f(n))$ on the running time.)

(3) Although the algorithm you analyzed in parts (1) and (2) is the most natural way to solve the problem - after all, it just iterates through the relevant entries of the array B. filling in a value for each - it contains some highly unnecessary sources of inefficiency. Give a different algorithm to solve this problem, with an asymptotically better running time. In other words, you should design an algorithm with running time $\mathcal{O}(g(n))$, where $\lim_{n\to\infty} \frac{g(n)}{f(n)} = 0$.

So in particular, you are required to:

i. *(10 Points)* Analyze the simple algorithm described there (steps (1) and (2)).

ii. *(10 Points)* Design a faster algorithm and analyze its running time (step (3)).

3. *(35 Points)* **Chapter 3 - Graphs.**

(a) *(10 Points)* **DFS vs. BFS** : Given a connected graph $G = (V, E)$, and a specific node $u \in V$. You run BFS from u, and you find the BFS tree T rooted at u. You run DFS from u, and you find the same DFS tree T rooted at u. Prove that $G = T$, i.e., G cannot contain any edges that do not belong to T.

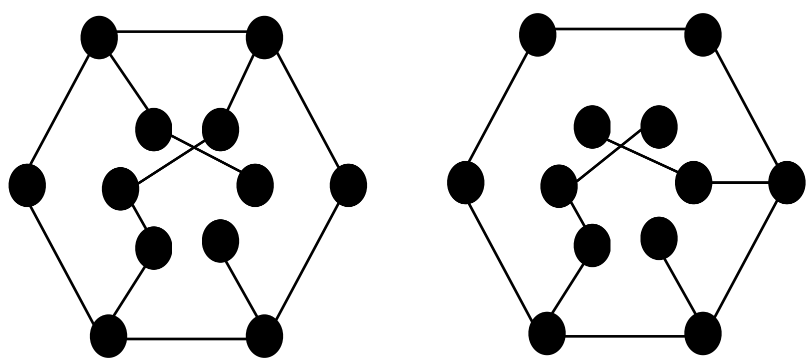
(b) *(10 Points)* **Connectivity** : Consider an undirected graph $G = (V, E)$ with $|V| = n$ nodes. Assume that there are two nodes s and t with distance between them strictly greater than $\frac{n}{2}$.

- Prove that there must exist some node v, different from s, t, such that deleting v from G disconnects s from t.
- Draw one example of G, s, t, v.
- Provide an algorithm that finds such a node v in $O(m + n)$ time.

(c) *(5 points)* You are visiting Irvine and you want to explore the city of Irvine and visit as much places as possible. You start your special tour from one place and you need to end your special tour at the same place. You want to minimize the effort so you don't want to visit the same place twice (except for the start point). Assume that the city of Irvine is represented as a directed graph with N node and M edges. Where a node is one place to visit and an edge is the route to take to this place. Your special tour in this graph starts at node u is a simple path that begins and ends at the same node u. Formally, a special tour is path $u, v_1, v_2, ..., v_i, ..., u$ where v_i are distinct and not equal to u for all i. **For every node in the given graph, tell whether it is possible for you to do your special tour starting at that node.**
Given: $N = 5$, $M = 6$
Graph represented as adjacency list as follows:

1	2	
2	3	
3	4	1
4	2	5
5		

(d) *(5 points)* Indicate which one of these two graphs is bipartite. Justify your answer.

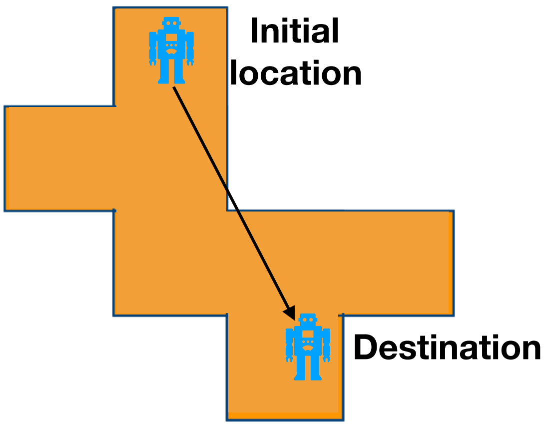


(e) *(5 points)* **Robotics Motion Planning**
In autonomous robot navigation problem, you want to design a controller that can navigate the robot within a bounded space. Assume that you have a localization algorithm running. Basically the robot can localize itself within this room. Assume you can discritize the space of your room into discrete number of squares. The robot does not know the structure of the room. Hence, you can not just draw a line between the initial position and the destination that the robot can follow. The visual capability of the robot can only let it see the neighboring discrete locations. Can you design a controller on the robot that can take the robot from one initial position to a destination position?

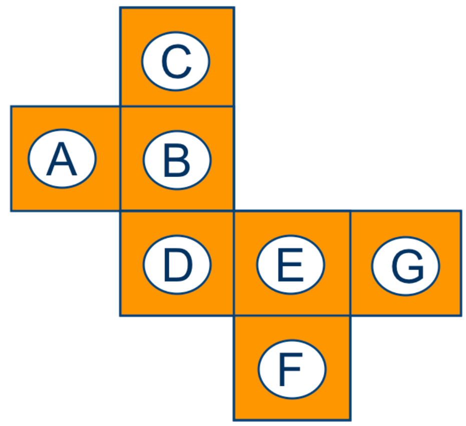
i. *(2 points)* Specify one of the algorithms that you learnt in the class to plan the motion of the robot.

ii. *(2 points)* Draw the route that the robot will take to reach its destination.

iii. *(1 points)* After the robot reached its destination (From C to F) as indicated in the discretized space below, the robot wants to go back to its initial location. Draw the path that the robot will take. What is the distance that the robot had to cover (in terms of discretized blocks) to return to its initial location?



Room with finite edges



Space can be discretized

Food for the thought! - No Credit

We studied the stable matching problem. A variant to it is when you have unequal number of men and women and the men and women only list who they are willing to marry to (not everyone and not in order of preference). Now suppose that the bipartite graph you indicated in question 3.e, if the blue set is the men and the red set is the women. The edge indicates a possible marriage between this man and this woman. How many compatible marriages are possible at the same time, and how can we find such an optimal matching?
In graph theory, this is called a **matching problem of maximum cardinality**. In the example 3.e, try to think of an algorithm to find this number; the maximum cardinality of the matching.