Problem Set 1

CMSC 27230: Honors Theory of Algorithms

Assigned January 6

Problem 1: Gale-Shapley and Stable Matchings (30 points)

(a) 10 points: Given the following preference lists, run the Gale-Shapley algorithm with group A making the offers to obtain a stable matching.

Group A's preference lists (from most preferred to least preferred):

a_1: b_1, b_4, b_3, b_2

a_2: b_1, b_2, b_4, b_3

a_3: b_4, b_1, b_2, b_3

a_4: b_4, b_1, b_3, b_2

Group B's preference lists (from most preferred to least preferred):

b_1: a_3, a_4, a_1, a_2

b_2: a_4, a_3, a_1, a_2

b_3: a_2, a_3, a_4, a_1

b_4: a_2, a_3, a_1, a_4

For your answer (and for your answer to part b as well), you should give the following data. At each step, give the offer that is made, whether or not this offer is accepted, and the resulting partial matching.

For this part (and for part b), you may write a computer program to implement the Gale-Shapley algorithm and generate the needed data. If you do so, you should submit your code along with the data.

(b) 10 points: Now run the Gale-Shapley algorithm with group B making the offers to obtain another stable matching. Which people are happier in this new stable matching (compared to the stable matching found in part a)?

(c) 10 points: What other stable matching(s) are there, if any? Note: For full credit you should show that you have indeed found all of the possible stable matchings. Hint is available.

Problem 2: Big O Notation (20 points)

(a) 10 points: Given each of the following statements, what can we say about $T(n)$ using Big O notation?

1. $T(n) = n^3 + 2n^2 + 20$
2. $T(n) \leq 10n + 3n(logn)^5$
3. $T(n) \geq 200log(n) + 10$
4. $\frac{2^n}{5} \leq T(n) \leq 7n^2 2^n$
5. $T(n) = n + (1 + (-1)^n)n^2 + 1000$

(b) 10 points: Put the following runtimes in order from smallest to largest:
$O(n)$, $O(2^n)$, $O((logn)^{10})$, $O(n^2)$, $O(log(n)^{log(n)})$, $O(nlog(n))$, $O(2^{\sqrt{log(n)}})$, $O(log(log(n)))$, $O(n^{\sqrt{n}})$, $O(\sqrt[3]{n})$, $O(1)$
Hint is available.

Problem 3: Can all matchings be stable? (10 points)

For which $n \in \mathbb{N}$ is it possible that all matchings between $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ are stable? Prove that your answer is correct.
Hint is available.

Problem 4: Manipulating Gale-Shapley (20 points)

Let's say that we are running the Gale-Shapley algorithm on behalf of two groups $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ of n people with group A making the offers. However, after everyone sends us their preference lists but before we run the algorithm, one of the people hacks into our system and reads all of the preference lists. This person then modifies his/her own preference list but leaves the other preference lists alone as touching someone else's preference list may be detected.

(a) 10 points: If the hacker a_i is in group A, is it possible that he/she can obtain a better partner by modifying his/her preference list? If so, give an example (i.e., the original preference lists and the modified preference list for a_i) where this happens. If not, prove that this cannot happen.

(b) 10 points: If the hacker b_j is in group B, is it possible that he/she can obtain a better partner by modifying his/her preference list? If so, give an example (i.e., the original preference lists and the modified preference list for b_j) where this happens. If not, prove that this cannot happen.

Note that with the modified preference list, the resulting matching may no longer be stable with respect to the original preference lists.

Problem 5: King of the Hill (20 points, 5 extra credit points available)

n people are playing the following variant of king of the hill:

1. Everyone starts at height $h = 1$.
2. Each round, the referee selects three people who are at the same height $h > 0$ (if possible) and has them compete to see who will go up and who will go down. The winner of the competition goes up to height $h + 1$ while the two losers go down to height $h - 1$. Anyone who goes down to height 0 is eliminated.
3. If there are at most two people at every height $h > 0$ then the game ends and the person or people at the highest height win.

(a) 10 points: Prove that the game will eventually terminate and give an upper bound on the number of competitions required before the game ends. (Hint is available)

(b) 10 points: How high up will the top person or people be? Express your answer using big O notation and justify your answer. (Hint is available)

Note: You do not need to give a rigorous proof for this part but you should have both a lower bound argument and an upper bound argument.

(c) Challenge question (5 points extra credit): After the game ends, exactly how many people will be at each height? Explain why your answer is correct.

Hints are available.

Hints

1c. Use parts a and b and the properties of the Gale-Shapley algorithm to determine the set of possible partners for each person in a stable matching and then work from there.

2b. Taking the logarithm of these runtimes may help.

3. Can you find a pair of people $a_i \in A$ and $b_j \in B$ such that b_j is not the last choice for a_i and a_i is not the last choice for b_j?

5a. Can you find a simple potential function which is guaranteed to decrease by 1 after each step?

5b. Can you find a potential function which is invariant after each step?