COMS21103: Data Structures and Algorithms

Problem Sheet - Week 9

1. **Stable Matching** For the Gale-Shapley algorithm formulation: n men $(m_1, m_2, ...m_n)$ and n women $(w_1, w_2, ...w_n)$ each with strict preference lists.

   (a) Find the stable matching for the preferences matrices below:

	1^{st}	2^{nd}	3^{rd}	4^{th}
Alex	Fiona	Emily	Gemma	Harriet
Bill	Gemma	Harriet	Fiona	Emily
Callum	Gemma	Fiona	Harriet	Emily
David	Emily	Harriet	Gemma	Fiona

	1^{st}	2^{nd}	3^{rd}	4^{th}
Emily	Callum	David	Bill	Alex
Fiona	Callum	Bill	Alex	David
Gemma	Alex	Bill	Callum	David
Harriet	David	Callum	Bill	Alex

   (b) Suppose that the boys all have different favourite girls. What would the complexity of the G-S algorithm be?

   $O(n)$ because all first proposals will be accepted.

   (c) Suppose that the boys have identical preferences. How long does it take for the algorithm to converge?

   Depends on the preferences of the girls, but could take up to $O(n^2)$

   (d) Consider the 'stable roommate' problem where n people rank each other in order of preference (assume n is even). The task is to pair up the people in a perfect matching as in the stable matching problem shown in lectures but this time there are no genders. That is anyone can match with anyone else. Give an example where there is no stable matching possible and explain why this is the case.

   Algorithm for stable roommate problem can be found here

   (e) Solve the following example using the 'stable roommate' problem to find a stable matching

A	:	D	F	B	E	C
B	:	F	C	E	A	D
C	:	D	E	A	F	B
D	:	B	F	E	A	C
E	:	D	B	C	F	A
F	:	E	A	D	B	C

   (f) *optional:* Use the code stubs platform to implement the Gale-Shapley algorithm.

   (g) *optional:* Use the code stubs platform to test your solution for the stable matching problem in (a).

2. **Order Statistics**

   (a) Let $X[1..n]$ and $Y[1..n]$ be two arrays, each containing n numbers already in sorted order. Give an $O(\log n)$ algorithm to find the median of all $2n$ elements in arrays X and Y.

   Hint: resulting array is always even $2n$ so the median is the average of two array elements.
   1) find median of both arrays m_x, m_y.
   2) if $m_x == m_y$ median is m_x
   3) if $m_x < m_y$, the median is present in $X[m_x, n], Y[1, m_y]$ inclusive

4) if $m_x > m_y$, the median is present in $X[1, m_x], Y[m_y, n]$ inclusive
4) repeat until only two elements are present in each subarray $[a, b]$, $[c, d]$
5) median = $\frac{\max(a,c)+\min(b,d)}{2}$

(b) For n distinct elements $x_1, x_2, .., x_n$ with positive weights $w_1, w_2, ..., w_n$ such that $\sum_{i=1}^{n} w_i = 1$, the weighted median is the element x_k satisfying $\sum_{x_i < x_k} w_i < \frac{1}{2}$ and $\sum_{x_i > x_k} w_i \geq \frac{1}{2}$. Show how to compute the weighted median of n elements in $\Theta(n)$.
Hint: in partitioning the array, keep a track of the weighted sum in at least one side