CMPT 307 (Fall 2013)
Quiz 1 (October 1, 2013)
Total marks = 40.

1. (15 points)

   (a) Consider the following preferences:

Girl	1	2	3	Boy	1	2	3
Allie	V	W	X	Von	C	B	A
Bobbie	X	W	V	Will	C	A	B
Cathy	V	X	W	Xander	C	B	A

   i. Apply Gale-Shapely (G-S) algorithm to find a stable matching.
      - **The stable matching you will get is** $(A, W), (B, X), (C, V)$**.**

   ii. Is $(A, V), (B, X), (C, W)$ a stable matching? Explain.
      - **No.** C **prefers** X **over** W**, and** X **prefers** C **over** B**.**

   (b) Decide whether you think the following statement is true or false. If it is true, given a short justification. If it is false, give a counterexample.

   In every instance of the stable matching problem, there is a stable matching containing a pair (m, w) such that m is ranked first on the preference list of w, and w is ranked first on the preference list of m.

   - **This is false. Consider the following preferences:**

```
m prefers w to w'
m' prefers w' to w
w prefers m' to m
w' prefers m to m'
```

     Note that there are no pairs satisfying the claim of the question.

   (c) (**Bonus**) Show that Gale-Shapely algorithm always produces a stable matching.
   - **Proved in page 10 of the text (item 1.7)**

2. (15 points)

   (a) Consider a set $S = \{12, 14, 5, 20, 8, 6, 17\}$.

   i. Build a max heap of S and store the heap in an array.

- **The elements in the array** $(20, 14, 17, 12, 8, 6, 5)$ **satisfy the heap property. This heap can be obtained in linear time. The method was discussed in the class. O(nlogn) solution is easy to get. This is done by inserting each element one by one into the heap and heapify after each insertion. Initially, the heap is assumed empty.**

ii. What is the height of the heap?

- **The height is** $\lfloor log_2 n \rfloor$**. In the example** $n = 7$

iii. How do you handle Extract-Max (deleting the maximum element in the heap) operation in a heap? What is the running time?

- **See page 62 of the text showing how ExtractMin is handled.**

(b) Indicate, for each pair of expressions (A, B) in the table below, whether A is O, Ω, Θ of B. Assume that k is a constant.

	A	B	O	Θ	Ω
a.	n	$nlogn$	yes	no	no
b.	$10logn$	$log(n^2)$	yes	yes	yes
c.	n^k	2^n	yes	no	no
d.	$n^{0.5}$	$4^{log_2 n}$	yes	no	no

Can show that $4^{log_2 n}$ **is** n^2**.**

(c) Suppose we want to evaluate the $p(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0$ using the following function. What is the worst case running time of the function?

```
Function Evaluate(x,n)
p = A[0];
xpower=1;
for i = 1 to n do
   xpower=x*xpower;
   p = p + A[i]*xpower;
end
```

- **The for loop operates** $O(n)$ **times. The number of multiplications in the evaluation is** $2n$**. However, we need only** $O(logn)$ **multiplications to evaluate all** $x^i, i = 1, 2, ..., n$**. This is shown below.**

(Bonus) Can you improve on this algorithm? (Just adjusting the constant in the growth rate is not enough.)

- **It is possible to evaluate $x^i, i = 1, 2, ..., n$ in $O(logn)$ time. We first compute and store $x^2, x^4, x^8, ..., x^{logn}$. We need $O(log_2 n)$ time since $x^{2i} = x^i \times x^i$. Thus given n, we look at the bit pattern of n. Suppose $n = 99$. Then the binary bit pattern of 99 is $< 1100011 >$. This means that $n = 1 \times 64 + 1 \times 32 + 0 \times 16 + 0 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$. Thus knowing $x^{2i}, i = 0, 1, 2, ...$, we compute $x^n = x^{64} + x^{32} + x^2 + x$ in $O(logn)$ time. In the question I was looking for this improvement.**

3. (10 points) Answer any two questions. Make sure that your answer is precise and complete.

   (a) Let G be an undirected graph on n nodes, where n is an even number. Show that if every node G has a degree at least $\frac{n}{2}$, the G is connected.

   - **The problem was given in homework 2.**

   (b) Problem on cycle detection.

   i. Describe an algorithm to determine if the undirected graph has a cycle. Is it optimal?

   - **The problem was discussed in the class.**

   ii. Describe an algorithm if a digraph has a cycle.

   - **The existence of a strongly connected component implies an existence of a cycle. One solution is to apply BFS from each node u to G and G^{rev}, and find the vertices reachable from the node u in G and G^{rev}. If there is a common vertex in these two lists, we can conclude that G has a directed cycle, and u lies on it. Otherewise repeat the process starting from another vertex. There is an elegant solution using the depth first search. We have not covered the material yet.**

   (c) Show that a connected graph G has an Euler tour if and only if the degree of each node is even.

   - **This was discussed in the class.**

   (d) If $G = (V, E)$ is an undirected graph, the complement of G is the graph $G' = (V, E')$ such that $(a, b) \in E'$ if and only if $(a, b) \notin E$. Informally, the complement of G is constructed by adding all possible edges to G and deleting the original edges of G.

   True or false? If G is connected, G' is disconnected. The answer must be supported by arguments.

- **The above claim is false. Consider** $G = (V, E)$ **to be a path of length** $|V| - 1$**. In this case** G' **is also connected.**