University of Waterloo
CS240 Spring 2016
Assignment 1

Due Date: Wednesday, May 18, at 5:00pm

Please read `http://www.student.cs.uwaterloo.ca/~cs240/s16/guidelines.pdf` for guidelines on submission. This assignment contains both written problems and a programming problem. Submit your written solutions electronically as a PDF with file name a01wp.pdf using MarkUs. We will also accept individual question files named a01q1w.pdf, a01q2w.pdf, ... , a01q5w.pdf if you wish to submit questions as you complete them.

Problem 5 contains a programming question; submit your solution electronically as a file named `teampq.cc`.

Note: you may assume all logarithms are base 2 logarithms: $\log = \log_2$.

Problem 1 [4+4+4+4=16 marks]

Provide a complete proof of the following statements from first principles (i.e., using the original definitions of order notation).

a) $27n^7 + 17n^3 \log n + 2016$ is $O(n^9)$

b) $n^2(\log n)^{1.0001}$ is $\Omega(n^2)$

c) $\frac{n^2}{n+\log n}$ is $\Theta(n)$

d) n^n is $\omega(n^{20})$

Problem 2 [4+4+4=12 marks]

For each pair of the following functions, fill in the correct asymptotic notation among Θ, o, and ω in the statement $f(n) \in \sqcup(g(n))$. Prove the relationship using any relationship or technique that described in class.

a) $f(n) = n^2 + 27n \log n + 2016$ versus $g(n) = n^2 \log n + 2016$

b) $f(n) = 10^n + 99n^{10}$ versus $g(n) = 75^n + 25n^{27}$

c) $f(n) = \sqrt{n}$ versus $g(n) = (\log n)^7$

Problem 3 [4+4+4+4=16 marks]

Prove or disprove each of the following statements. To prove a statement, you should provide a formal proof that is based on the definitions of the order notations. To disprove a statement, you can either provide a counter example and explain it or provide a formal proof. All functions are positive functions.

a) $f(n) \notin o(g(n))$ and $f(n) \notin \omega(g(n)) \Rightarrow f(n) \in \Theta(g(n))$

b) $f(n) \in \Theta(g(n))$ and $h(n) \in \Theta(g(n)) \Rightarrow \frac{f(n)}{h(n)} \in \Theta(1)$

c) $f(n) \in \Theta(g(n)) \Rightarrow 2^{f(n)} \in \Theta(2^{g(n)})$

d) $\min(f(n), g(n)) \in \Theta\left(\frac{f(n)g(n)}{f(n)+g(n)}\right)$

Problem 4 [4+4+4=12 marks]

Analyze the following piece of pseudocode and give a tight (Θ) bound on the running time as a function of n. Show your work. A formal proof is not required, but you should justify your answer (in all cases, n is assumed to be a positive integer).

a)
```
x = 0
for i = 1 to n + 12 do
   x = x * 4
   for j = 389 to 20100
      for k = 2i to 3i
         x = x * 77
```

b)
```
x = 0
for i = 1 to ceiling(log(n))
   for j = 1 to i
      for k = 1 to 10
         x = x + 1
```

c)
```
x = 0
for i = 1 to sqr(n)    \\ i.e. n^2
   for j = 1 to ceiling(log(i))
      x = x + 1
```

Problem 5 [3+5+10 marks]

a) Show that an *arbitrary* item can be removed from a heap in $O(\log n)$ time, if the item's index into the heap is known. Specifically, give an algorithm *heapRemove(h, i)* that removes the element at index i from heap h. Justify your algorithm's running time.

b) Consider the following C++ class:

```
class Team {
   int wins;
   int losses;
   string name;
   // You may add fields/methods/constructors/destructor as necessary
};
```

We wish to build a data structure that allows for efficient retrieval of the team with the most wins and also for efficient retrieval of the team with the fewest losses. Essentally we want a priority queue that supports two different priority measures. Provide an implemenation of the following class:

```
class TeamPQ {
   // add fields/methods/constructors/destructor as necessary
 public:
   void insert(const Team &t);   // O(log n) time

   const Team &findMaxWins() const;    // O(1) time
   const Team &findMinLosses() const;  // O(1) time

   void removeMaxWins();     // O(log n) time
   void removeMinLosses();   // O(log n) time
};
```

Give pseudocode implementations of these routines, justify the runtimes, and submit with your written submission. Then implement this class in C++ and submit electronically (the remainder of these instructions pertain to the programming portion of this assignment).

Provide a main function that accepts the following commands from stdin (you may assume that all inputs are valid):

- `i wins losses name` - inserts a team with the given wins, losses, and name into the priority queue. You may assume that `name` contains no whitespace.
- `pw` - prints the name of the team with the most wins, without removing it from the priority queue. Prints nothing if the priority queue is empty.

- `pl` - prints the name of the team with the fewest losses, without removing it from the priority queue. Prints nothing if the priority queu is empty.
- `rw` - removes the team with the most wins from the priority queue and prints nothing. Does nothing if the priority queue is empty.
- `rl` - removes the team with the fewest losses from the priority queue and prints nothing. Does nothing if the priority queue is empty.

All output is printed to stdout. The program terminates when eof is encountered.

For example, the following input:

```
i 15 5 jays
i 13 3 yankees
i 12 7 orioles
i 10 10 rays
i 0 14 sox
pw
pl
rw
pw
rl
pl
i 20 1 jays
pw
pl
```

should produce the following output:

```
jays
yankees
yankees
orioles
jays
jays
```

Place your entire program in the file `teampq.cc`