CS3383, Winter 2019, Midterm Preparation

Rasoul Shahsavarifar
Faculty of Computer Science, UNB

Referring to the CS3383 calendar, the midterm will be held in class, Feb/25/2019.

Notes:

(A) The test will be closed book.

(B) All materials already on D2L including Lecture Slides, Notes, Assignments, Exercises, and the examples in the Textbook are supposed to be studied for the midterm.

(C) In addition to the materials mentioned in (B), some sample questions including the questions from from Dr. Bremner's algorithm page (`http://www.cs.unb.ca/~bremner/teaching/cs3383/tests/t1/`) are as follows:

1. Prove the following asymptotic bounds:

   a) $\forall\ a > 1\ ;\ an \in O(n^a)$

   b) $\forall\ a > 0\,, b > 0\ ;\ n^a + n^b \in O(n^c)$, where $c = a + b$

2. Solve the recurence $T(n) \leq 3T(n/4) + O(\sqrt{n})$.

3. Suppose $T(n) = T(n/2) + \log(n)$. Prove that $T(n) \in O(\log n)^2$.

4. Consider the following recursive algorithm to append two arrays.

$$
\begin{array}{l}
\textit{Function}\ \ \textit{Append}(A[1 \dots k]\ ,\ B[1 \dots l]) \\
\quad \textit{if}\ \ k = 0 \\
\quad\quad \textit{return}\ \ B \\
\quad \textit{if}\ \ l = 0 \\
\quad\quad \textit{return}\ \ A \\
\quad C = \textit{Append}(A[1 \dots k]\ ,\ B[1 \dots l-1]) \\
\quad \textit{return}\ \ C \circ B[l] \\
\textit{End}
\end{array}
$$

   The symbol $\circ$ represents the operation of appending one element to an array. Analyze this algorithm under the assumption that for an array S of size p and element e, the cost of computing $S \circ e$ is

   a) $\Theta(1)$

   b) $\Theta(p)$

5. Suppose that Unlucky Pete has probability p of passing $STAT1001$, no matter how many times he has taken it before. Let N be the random variable that counts the total number of times Pete takes $STAT1001$.
   Express N as a sum of indicator random variables. Use this sum to compute the expected value of N.

6. Compute the expected running time of the following algorithm.

$$
\begin{array}{l}
Function\ \ MAX(A[1 \dots n]) \\
repeat \\
\quad p \leftarrow random\ \ integer\ \ from\ \ 1 \dots n \\
\quad j \leftarrow 0 \\
for\ \ i = 1 \dots\ \ do \\
\quad if\ \ A[i] \geq A[p]\ \ then \\
\qquad j \leftarrow j + 1 \\
until\ \ j = 1 \\
\quad return\ \ A[p] \\
End
\end{array}
$$

7. Solve the following recurrences
   a) $T(n) = T(n-1) + n^c$
   b) $T(n) = T(n-1) + c^n$
   c) $T(n) = \sqrt{n}\, T(\sqrt{n}) + n$
   d) $T(n) = T(3n/4) + n$
   e) $T(n) = 2T(n/2) + n/(\log n)$
   f) $T(n) = 4T(n/2) + n \log n$

8. Let e be the unique lightest edge in a graph G. Let T be a spanning tree of G such that $e \notin T$. Prove using elementary properties of spanning trees (i.e. not the cut property) that T is not a minimum spanning tree of G.

9. A bridge is an edge whose removal disconnects the graph. Prove that any bridge must be in some minimum spanning tree. You may use the cut property in the proof, if you want.

10. Think about verification algorithms for NP-Compete problems such as: N-Queen, SudoKu, SAT, Rubik's Cube,...