COMP108 Algorithmic Foundations — Tutorial 7

w/c 20th March 2017

Name: ____

Hand in your answer for the question(s) marked "Do this during tutorial" to the demonstrator at the end of the tutorial (even if you haven't finished it). You will get feedback in the next tutorial. Tutorial participation contributes to 5% of overall marks.

1. Consider the following recurrence

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n-1) + 1 & \text{if } n > 1 \end{cases}$$

   Show by the iterative method that $T(n) = n$.

$$\begin{aligned}
T(n) &= T(n-1) + 1 && \leftarrow \text{ use the recurrence for } T(n) \\
&= && \leftarrow \text{ use recurrence for } T(n-1) \\
&= && \leftarrow \text{ expand the bracket} \\
&= && \leftarrow \text{ use recurrence for } T(n-2) \\
&= && \leftarrow \text{ expand the bracket} \\
&= && \leftarrow \text{ use recurrence for } T(n-3) \\
&\vdots \\
&= && \leftarrow \text{ in terms of } T(1) \\
&= && \leftarrow \text{ further simplification}
\end{aligned}$$

2. **[Do this before tutorial]** In Tutorial 6, you are asked to write the following pseudo code. If you haven't done this yet, try it before the tutorial.

   Describe or write pseudo code for a divide-and-conquer algorithm to find the product of the numbers in an array $A[\,]$ with n integers $A[1]..A[n]$.

   For simplicity, you can assume that n is a power of 2.

3. **[Do this during tutorial]** The time complexity $T(n)$ of your algorithm in Q.2 can be described by the following recurrence.

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 2 \times T(\frac{n}{2}) + 1 & \text{if } n > 1 \end{cases}$$

Solve the recurrence by the iterative method. *(Hint 1: prove that $T(n) = 2n - 1$.) (Hint 2: you can use the facts that (i) $n = 2^{\log n}$ and (ii) $1 + 2 + 2^2 + 2^3 + 2^4 + \cdots + 2^{\log n} = 2^{\log n + 1} - 1$.)*

$T(n)$	$=$	$2 \times T(\frac{n}{2}) + 1$	$\leftarrow$ use the recurrence for $T(n)$
	$=$	$2 \times (\qquad\qquad\qquad) + 1$	$\leftarrow$ use recurrence for $T(\frac{n}{2})$
	$=$		$\leftarrow$ expand the bracket
	$=$		$\leftarrow$ use recurrence for $T(\frac{n}{2^2})$
	$=$		$\leftarrow$ expand the bracket
	$=$		$\leftarrow$ use recurrence for $T(\frac{n}{2^3})$
		$\vdots$	
	$=$		$\leftarrow$ in terms of $T\left(\frac{n}{2^{\log n}}\right)$, i.e., $T(1)$
	$=$		$\leftarrow$ further simplification

4. **[Do this after tutorial]** Referring to the recurrence in Q.3, prove that $T(n) = 2n - 1$ using the substitution method (i.e., by Mathematical Induction).

5. **[Puzzle]** An 8×8 chessboard has had two of its diagonally opposite squares removed, leaving it with sixty-two squares. It is said that we cannot tile the chessboard with 31 non-overlapping 2×1 rectangles (dominoes). Do you agree with it? Why?