# CSE 3101: Assignment 1

Worth 10%. Due May 16 at the beginning of class.

*The work you submit must be your own.* You may discuss problems with each others; however, you should prepare written solutions alone. Copying assignments is a serious academic offence, and will be dealt with accordingly.

**Question 1** Let $c$ be a positive constant, $c \neq 1$. Prove the following equalities by induction on $n$, for $n \geq 1$. (Note that when $n = 1$, the LHS of both equalities are just 1.)

**a)**

$$1 + c + c^2 + \ldots + c^{n-1} = \frac{c^n - 1}{c - 1}$$

**b)**

$$1 + 2c + 3c^2 + \ldots + nc^{n-1} = \frac{nc^{n+1} - (n+1)c^n + 1}{(c-1)^2}$$

**Question 2** The following recurrence often arises in analyzing the time complexity of recursive algorithms:

$$S(n) = aS(n-1) + g(n) \qquad \text{for } n \geq 2 \tag{1}$$

where $a > 0$ and $S(1) > 0$ are some constants.

In parts **a)**, **b)** and **c)** below we consider the case where $g(n) = d^n$, for some constant $d > 0$. Then we will consider the case where $g(n) = cna^n$ in parts **d)** and **e)**.

First, suppose that $g(n) = d^n$, for some constant $d > 0$. We can "unwind" $S(n)$ as follows:

$$\begin{aligned} S(n) &= aS(n-1) + d^n \\ &= a(aS(n-2) + d^{n-1}) + d^n \\ &= a^2S(n-2) + [ad^{n-1} + d^n] \\ &= a^2(aS(n-3) + d^{n-2}) + [ad^{n-1} + d^n] \\ &= a^3S(n-3) + [a^2d^{n-2} + ad^{n-1} + d^n] \\ &\ldots \\ &= a^{n-1}S(1) + [a^{n-2}d^2 + a^{n-3}d^3 + \ldots + ad^{n-1} + d^n] \end{aligned} \tag{2}$$

**a)** When $d = a$, show that $S(n) = \Theta(na^n)$.

Now consider the case where $d \neq a$. Write (2) as

$$S(n) = a^{n-1}S(1) + a^{n-2}d^2 \left[ 1 + \frac{d}{a} + \left(\frac{d}{a}\right)^2 + \ldots + \left(\frac{d}{a}\right)^{n-2} \right]$$

Use the result from Question 1 **a)** to show that

**b)** If $a > d$ then $S(n) = \Theta(a^n)$.
**c)** If $d > a$ then $S(n) = \Theta(d^n)$.

Now consider the recurrence (1) where $g(n) = cna^n$, for some constant $c > 0$.

**d)** Expand $S(n)$ in the style of (2).
**e)** Show that $S(n) = \Theta(n^2 a^n)$.

**Question 3** In this question you are asked to come up with an array of 11 natural numbers which is somewhat specific to you, and then run the Heapsort algorithm on it. Consider the following array $A$ with missing values for $A[1], A[2], A[7], A[8], A[11]$.

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 20 | 6 | 16 | 5 | | | 31 | 12 | |
| A | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |

**a)** Write down the complete array $A$ where

- $A[1]$ and $A[2]$ are the day and month of the birthday of one close friend of yours.
- $A[7]$ is the sum of the digits in your student number.
- $A[8]$ is the sum of the digits in your year of birth.
- $A[11]$ is the sum of the numbers for the day and month of your birthday.

**b)** Draw the binary tree with 11 nodes, whose nodes are labeled with $A[1], \ldots, A[11]$, so that the two childrens of the node $A[i]$ are nodes $A[2i]$ and $A[2i+1]$, for $1 \leq i \leq 5$.

**c)** Now we follows the Build-Heap procedure on $A$. Show the run of Build-Heap on $A$ by drawing, for each $i \leq 5$, the subtree at $A[i]$ as it is being modified during the call to $\textit{Heapify}(A, i)$.

**d)** Write down the updated array $A$. Exchange $A[1]$ and $A[11]$. Write down the resulted array.

**e)** Follow the Heapsort algorithm. Now the new heap-size is 10, and we need to consider only the sub-array $A[1], \ldots, A[10]$.

- **(i)** Draws the new heap (of size 10).
- **(ii)** Show how the heap changes during the run of $\textit{Heapify}(A, 1)$.
- **(iii)** Write down the updated array $A$.