College of Computer & Information Science	CS4800
Northeastern University	Algorithms
Fall 2017	September 12 2017

Algorithms - CS4800 - Fall 2017 - Assignment 1 - Solution
(Due Tuesday, September 19)

The assignment should be submitted through Blackboard by 11:59:00. We prefer typed submissions but we will accept hand written work provided it is neatly presented.

1. (10 points) Consider the following 12 functions for the question that follows:

   (a) $\log_2(n)$
   (b) $10 \cdot n!$
   (c) n^n
   (d) $500 \cdot 2^n$
   (e) $\frac{1}{312}n^6$
   (f) $25 \cdot n^2$
   (g) $\sqrt{n}$
   (h) $n \log_2(n)$
   (i) $\log_{10}(n^2)$
   (j) $1 + 2 + ... + n$
   (k) $|1 - 2 + 4 - 8 + ... + (-1)^n \cdot 2^n|$
   (l) $(n - 1)!$

   Make a table in which each function is in a column dictated by its big-Θ growth rate. Functions with the same asymptotic growth rate should be in the same column. Columns should be ordered left to right by the rate of growth of their functions: columns with slower growing functions should be to the left of columns with faster growing functions.

   Solution:

$\log_2(n)$ $\log_{10}(n^2)$	$\sqrt{n}$	$n \log_2 n$	$25n^2$ $1 + 2 + ... + n$	$\frac{1}{312}n^6$	$500 \cdot 2^n$ $\lvert 1 - 2 + 4 - 8 + ... + (-1)^n \cdot 2^n \rvert$	$(n-1)!$	$10 \cdot n!$	n^n

2. (10 points) For each of the following functions $f(n)$, find a simple function $g(n)$ such that $f(n) \in \Theta(g(n))$.

   a. $f(n) = (n^2 + 1)^{10}$
   b. $f(n) = \sqrt{10n^2 + 7n + 3}$
   c. $f(n) = 2n \log_2[(n + 2)^2] + (n + 2)^2 \log_2(\frac{n}{2})$
   d. $f(n) = 2^{n+1} + 3^{n-1}$
   e. $f(n) = \lfloor \log_2 n \rfloor$

   Solution:

   a. $g(n) = n^{20}$
   b. $g(n) = n$
   c. $g(n) = n^2 \log_2(n)$
   d. $g(n) = 3^n$
   e. $g(n) = \log_2 n$

Explanations:

(a) For any $n \geq 1$ we have:

$$n^{20} = (n^2)^{10} \leq (n^2+1)^{10} \leq (n^2+n^2)^{10} = 1024n^{20}$$

(b) By properties of the logarithmic function, removal of constants and lower order terms of the function:

$$2n\log_2[(n+2)^2] + (n+2)^2\log_2\left(\frac{n}{2}\right)$$

one get that it is $\Theta(n^2\log_2 n)$.

(c) For any $n \geq 1$ we have:

$$n = \sqrt{n^2} \leq \sqrt{10n^2+7n+3} \leq \sqrt{10n^2+7n^2+3n^2} = \sqrt{20}\cdot n$$

(d) For any $n \geq 1$ we have:

$$\frac{1}{3}\cdot 3^n = 3^{n-1} \leq 2^{n+1} + 3^{n-1} \leq 3^{n+1} + 3^{n+1} = 6\cdot 3^n$$

3. (10 points) Let $f(n)$ and $g(n)$ be non-negative functions. Prove or disprove:

$$\max\{f(n), g(n)\} = \Theta(f(n)+g(n))$$

<u>Solution:</u>
The proposition is correct.
Proof: Clearly, for any $n \in \mathbb{N}$ we have:

$$\max\{f(n), g(n)\} \leq (f(n)+g(n))$$

It is also clear that for any $n \in \mathbb{N}$ we have:

$$(f(n)+g(n)) \leq \max\{f(n), g(n)\} + \max\{f(n), g(n)\} = 2\max\{f(n), g(n)\}$$

We see that for any $n \in \mathbb{N}$ we have:

$$\frac{1}{2}(f(n)+g(n)) \leq \max\{f(n), g(n)\} \leq (f(n)+g(n))$$

4. (10 points) You were driving your car on a straight road from east to west in the dessert when at some point it broke down. You know that there is only one well on the road but you do not know how far the well is from where you are, and whether it is to the east or to the west.
Describe a $O(n)$ algorithm to find the well, where n is the distance from your location to the well. The efficiency of your algorithm is measured in terms of

the number $T(n)$ of miles you travel until you find the well.

Solution:
The algorithm is:
Walk one mile right, then 2 miles to the left, then 4 to the right, and so on.
After k turns:

$$\text{The distance traveled} = 1 + 2 + 2^2 + ... + 2^k = 2^{k+1} - 1$$

$$\text{Location reached} = 1 - 2 + 2^2 - ... + (-2)^k = \frac{(-2)^{k+1} - 1}{-3}$$

The formula implies that for k odd we have Distance $= 3|\text{Location}|$ and for k even we have Distance$+2 = 3|\text{Location}|$. It follows that Distance $\leq 3|\text{Location}|$.

More formally, if you wish:
The algorithm is:
1. k=0
2. While you didn't find the well
3. if k is even
4. go 2^k miles to the west
5. else
6. go 2^k miles to the east
7. k=k+1

5. (10 points) Use forward substitutions to solve the following recurrences.

   (a)

$$T(n) = \begin{cases} 1, & \text{n=1.} \\ 3T(n-1) + 2, & n > 1; \end{cases}$$

   (b) solve for powers of 3 only:

$$T(n) = \begin{cases} 1, & \text{n=1.} \\ 3T\left(\frac{n}{3}\right) + 1, & n > 1; \end{cases}$$

Solution:

   (a) We have:
   $T(1) = 1$
   $T(2) = 3 \cdot 1 + 2$
   $T(3) = 3(3 + 2) + 2 = 3^2 + 3 \cdot 2 + 2$
   $T(4) = 3(3^2 + 3 \cdot 2 + 2) + 2 = 3^3 + 3^2 \cdot 2 + 3 \cdot 2 + 2$
   Generally, we get:

$$T(n) = 3^{n-1} + 2(3^{n-2} + 3^{n-3} + ... + 1) = 3^{n-1} + 2 \cdot \frac{3^{n-1} - 1}{3 - 1} = 2 \cdot 3^{n-1} - 1$$

   We can prove this by induction:

$$T(n+1) = 3T(n) + 2 = 3 \cdot (2 \cdot 3^{n-1} - 1) + 2 = 2 \cdot 3^n - 1$$

(b) We have:
$T(1) = 1$
$T(3) = 3 \cdot 1 + 1$
$T(3^2) = 3(3+1) + 1 = 3^2 + 3 + 1$
$T(3^3) = 3(3^2 + 3 + 1) + 1 = 3^3 + 3^2 + 3 + 1$
Generally, we get:

$$T(3^k) = 3^k + 3^{k-1} + ... + 3 + 1 = \sum_{m=0}^{k} 3^m = \frac{3^{k+1} - 1}{3 - 1} = \frac{3^{k+1} - 1}{2}$$

So for $n = 3^k$ we have:

$$T(n) = \frac{3n - 1}{2}$$

One should verify that this satisfies the recurrence.

6. (10 points) Find an explicit expression for the function:

$$T(n) = \begin{cases} 1, & n = 1; \\ T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + 1, & n > 1 \end{cases}$$

First solve the recurrence for powers of 2 then use induction to prove that your formula is valid for all n.

<u>Solution:</u>
We have:
$T(1) = 1$
$T(2) = T(1) + T(1) + 1 = 3$
$T(3) = T(1) + T(2) + 1 = 1 + 3 + 1 = 5$
$T(4) = T(2) + T(2) + 1 = 3 + 3 + 1 = 7$
$T(5) = T(2) + T(3) + 1 = 3 + 5 + 1 = 9$
Generally, we can guess that:

$$T(n) = 2n - 1$$

We can prove this by strong induction.
The base case $T(1) = 1$ is trivial.
Suppose now that $n > 1$ and that $T(k) = 2k - 1$ for all $k < n$.
By definition we have:

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + 1$$

Since $\lfloor \frac{n}{2} \rfloor < n$ and $\lceil \frac{n}{2} \rceil < n$ for any $n \geq 1$, we can use the induction hypothesis that $T(k) = 2k - 1$ for $k < n$:

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil\right) + 1 = 2\left\lfloor \frac{n}{2} \right\rfloor - 1 + 2\left\lceil \frac{n}{2} \right\rceil - 1 + 1 = 2n - 1$$

■