CMPT 307 : Quiz 1 (Total Marks: 40) Time: 50 minutes
October 3, 2019

Answer questions 5 and 6, and any 15 points from the rest.

1. (5 points) Consider a LinkedList that has a **get**(int i) method to return the i-th element in the list. Write a pseudocode to implement the method. Provide the analysis (in order notation, in terms of n).

   Ans:

```
get(L_head,i):

*ptr = L_head;
j=1;
while j < i do  {
        *ptr = ptr->next;
        j++
}
return ptr->key;
```

   Clearly, the cost of retrieving the ith element is $O(i)$.

2. (5 points) Group the following time functions by efficiency class. Show your work.

   (a) $f(n) = 6\log_2 n$ $\quad f(n) \in \Theta(\log n)$
   (b) $f(n) = 5n^3$ $\quad f(n) \in \Theta(n^3)$
   (c) $f(n) = \frac{n}{4}$ $\quad f(n) \in \Theta(n)$
   (d) $f(n) = 2^n$ $\quad f(n) \in \Theta(2^n)$
   (e) $f(n) = 10n\log_2 n + 100$ $\quad f(n) \in \Theta(n\log n)$

   Now we can order the efficiency class as follows:

$$\Theta(\log n) \subset \Theta(n) \subset \Theta(n\log n) \subset \Theta(n^3) \subset \Theta(2^n)$$

3. (5 points) Apply Master Theorem to determine the closed form of the following recurrence relations. You can assume that for each of the following the basis is $T(1) = 1$.

(a) $T(n) = 4T(n/2) + n\log^4 n,\ n \geq 2$

Ans: From the Master Theorem formulation we notice that $a = 4; b = 2$. Also $n\log^4 n \in \Omega(n^1)$ and $n\log^4 n \in O(n^{1+\varepsilon})$. Therefore, $d = 1$. Now $\frac{a}{b^\alpha} = 1$, i.e $\frac{4}{2^\alpha} = 1$ implies $\alpha = 2$. Since $\alpha < d$, the splitting cost dominates. Therefore $T(n) \in O(n^\alpha)$, i.e. $T(n) \in O(n^2)$.

(b) $T(n) = 2T(n/2) + n\log n,\ n \geq 2$

Ans: From the Master Theorem formulation we notice that $a = 2; b = 2$. Also $n\log n \in \Omega(n^1)$ and $n\log n \in O(n^{1+\varepsilon})$. Therefore, $d = 1$. Now $\frac{a}{b^\alpha} = 1$, i.e $\frac{2}{2^\alpha} = 1$ implies $\alpha = 1$. Since $\alpha = d$, $T(n) \in O(n\log^2 n)$.

4. (10 points) Consider the following recurrence relation

$$\begin{aligned} T(1) &= O(1) \\ T(n) &= 2T(n/2) + n^2,\ \ n \geq 2. \end{aligned}$$

(a) Draw the recursion tree and determine its height.

Ans: The recursion tree has branching factor 2. The height of the tree is $\log_2 n$. I am assuming that n is a perfect power of 2.

(b) How many nodes are there on the i^{th} level of the recursion tree?

At level 0 (the root level), there is one node. At level 1 the number of nodes is 2. At the i^{th} level, there are 2^i nodes.

(c) What is the size of each subproblem on the i^{th} level of the recursion tree?

Ans: The size of each subproblem is $\frac{n}{2^i}$.

(d) What is the total merging cost of the subproblems on the i^{th} level of the recursion tree.

Ans: Total cost is $2^i \times (\frac{n}{2^i})^2$ which is $\frac{n^2}{2^i}$.

(e) Determine the closed form of $T(n)$.

Ans: Here, the merging cost dominates. Check it using the Master Theorem. Total merging cost is

$$\sum_{i=0}^{\log_2 n} \frac{n^2}{2^i}.$$

This is a geometric series with common ratio $\frac{1}{2}$ which is less than 1. Therefore, $T(n) = \Theta(n^2)$, the first term dominates the sum of the rest of the terms.

5. (10 points) The following questions are selected from the "why?" list in the study guide.

   (a) Show that $a^{log_b n} = n^{\log_b a}$
   Ans: Taking logarithm to the base b to the both sides of $a^{log_b n} = n^{\log_b a}$ we get $\log_b n \times \log_b a = \log_b a \times \log_b n$.

   (b) Let x_L and x_R be the left and the right $n/2$ bits respectively of an n-bit integer x where n is a perfect power of 2. Similarly, let y_L and y_R be the left and the right $n/2$ bits respectively of an n-bit integer y. Show that
   $$xy = 2^n x_L y_L + 2^{n/2}(x_L y_R + x_R y_L) + x_R y_R.$$
   Ans: Described in the text page 46.

   (c) Show that $f(n)\log_b n \in \Theta(f(n)\log_2 n)$ for any $f(n)$ and constant integer $b > 1$.
   Ans: We know that $\log_b n = \frac{\log_2 n}{log_2 b}$ for any $b > 1$. Now
   $$lim_{n\to\infty}\frac{f(n)\log_b n}{f(n)\log_2 n} = lim_{n\to\infty}\frac{1}{log_2 b} = \frac{1}{log_2 b},\ a\ constant.$$
   Therefore, $f(n)\log_b n \in \Theta(f(n)\log_2 n)$.

6. (a) (15 points) Describe Euclid's algorithm to compute the greatest common divisor of positive integers M and N of n binary bits long. Show that the algorithm runs in polynomial time.

   Ans: The algorithm is described in Fig 1.5 of the text.

   The algorithm is a recursive one. As a consequence of the lemma in page 23, after two consecutive rounds, both arguments, a and b, are at the very least halved in value. Therefore, the recursion tree of Euclid(a,b) has height at most $2n$. Every level involves one division. We just compute the total operation cost at all the even levels; level 0, level 2,...,level $2n$. We then multiply by 2 to get a bound on the total operation cost in all the levels. At level $2i$ the division costs at most

$(n-i)^2$ bit operations since both the operands have at most $n-i$ bits. Therefore, the total bit operations at all the levels is

$$2[(n-1)^2+(n-2)^2+(n-2)^2+\ldots+(n-1)^2].$$

Thus the total bit complexity of Euclid's algorithm is $O(n^3)$.

(b) Find the greatest common divisor d of 1102 and 399.

$$\begin{array}{lcll} 1102 & = & 2\times 399+304 & \text{.... (1)} \\ 399 & = & 1\times 304+95 & \text{.... (2)} \\ 304 & = & 3\times 95+19 & \text{.... (3)} \\ 95 & = & 5\times 19+0 & \text{.... (4)} \end{array}$$

Therefore, $gcd(1102,399)=19$.

(c) Find the integers x and y, in the solution of $1102x+399y=d$. (Just reverse the steps in (b).)
Answer: We can write

$$19=304-3\times 95(eq.3)=304-3\times(399-1\times 304)(eq.2)=4\times 304-3\times 399$$

$$i.e.19=4\times(1102-2\times 399)(eq.1)-3\times 399=4\times 1102-11\times 399$$

(d) **(Bonus Question)** Determine the smallest integer n which can be expressed as $n=1102\alpha+399\beta$ where $\alpha,\beta\geq 0$.
Ans: It is trivial to see that $\alpha=\beta=0$ will give $n=0$.