CSC236 Tutorial 6

Sample Solutions

1. Let $T(n)$ denote the worst-case running time of the algorithm below on inputs of size n.

```
# A is a list.
   def fun(A):
1.     if len(A) < 2:
2.         return 1
3.     else:
4.         m = len(A)//2 # Integer division
5.         return fun(A[0..(m − 1)]) ∗ fun(A[m..(len(A) − 1)])
```

   (a) Write a recurrence relation satisfied by T. You may assume that $len(A)$ is a power of 2.
   Make sure to define n precisely (as a function of the algorithm's parameters) and justify that your recurrence is correct (by referring to the algorithm to describe how you obtained each term in your answer).

   (b) Give an asymptotic upper-bound for the worst-case running time of the algorithm.

   Solutions:

   (a) For any natural number n, let $T(n)$ denote the maximum number of steps executed by a call to $fun(A)$, where $n = len(A)$.
   If $n < 2$, then lines 1 and 2 execute, which take constant time, represented by a constant value a. Otherwise, lines 3 to 5 execute. There are two recursive calls in line 5, each on a list of size $\frac{n}{2}$. Therefore, lines 5 takes $2T(\frac{n}{2})$.
   All other instructions in lines 3–5 take constant time, represented by a constant value b. Putting all together, we get the following definition for $T(n)$:

$$T(n) = \begin{cases} a, & n = 0 \text{ or } n = 1 \\ 2T(\frac{n}{2}) + b, & n \geq 2 \end{cases}$$

   (b) We will use the Master Theorem.

   Here, $c = 2, d = 2$, and $k = 0$. Since $\log_2 2 = 1 > k$, by the Master Theorem, $T(n) \in \mathcal{O}(n)$.

2. Let $a, b \in \mathbb{N}$. Consider the following function $f : \mathbb{N} \to \mathbb{N}$.

$$f(n) = \begin{cases} 4, & n = 1 \\ f(\frac{n}{3}) + n^2 - 3, & n \geq 2 \end{cases}$$

   Find a closed-form expression for f. You may assume that n is a power of 3.
   You don't need to prove the correctness of the closed-form expression you obtained.

Solution: Assume $n \geq 2$. Then

$$\begin{aligned}
f(n) &= f(\frac{n}{3}) + n^2 - 3 \\
&= [f(\frac{n}{3^2}) + (\frac{n}{3})^2 - 3] + n^2 - 3 \\
&= f(\frac{n}{3^2}) + n^2(\frac{1}{9} + 1) - 6 \\
&= [f(\frac{n}{3^3}) + (\frac{n}{3})^3 - 3] + n^2(1 + \frac{1}{9}) - 6 \\
&= f(\frac{n}{3^3}) + n^2(\frac{1}{81} + \frac{1}{9} + 1) - 9
\end{aligned}$$

It seems that after i applications of the recursive definition we have

$$f(n) = f(\frac{n}{3^i}) + n^2(\frac{1}{9^{i-1}} + \frac{1}{9^{i-2}} + ... + \frac{1}{9^0}) - 3i$$

Therefore, after $\log_3 n$ applications of the recursive definition we have (note that $f(1) = 4$)

$$\begin{aligned}
f(n) &= f(\frac{n}{n}) + n^2 \sum_{t=0}^{\log_3 n - 1} \frac{1}{9^t} - 3\log_3 n && (1) \\
&= 4 + n^2 \sum_{t=0}^{\log_3 n - 1} \frac{1}{9^t} - 3\log_3 n && (2)
\end{aligned}$$

We use the geometric series formula to get:

$$\sum_{t=0}^{\log_3 n - 1} \frac{1}{9^t} = \frac{1 - \frac{1}{9^{\log_3 n}}}{1 - \frac{1}{9}} = \frac{9}{8}(1 - \frac{1}{n^2})$$

(Note that $9^{\log_3 n} = (3^2)^{\log_3 n} = (3^{\log_3 n})^2 = n^2$).

Substituting this in (2) yields the following closed-form expression:

$$\begin{aligned}
f(n) &= 4 + \frac{9}{8}n^2 - \frac{9}{8} - 3\log_3 n \\
&= \frac{9}{8}n^2 - 3\log_3 n + \frac{23}{8}
\end{aligned}$$