EC 504
Spring, 2021
Quiz 1 Solutions

1. Order the following functions in increasing order of asymptotic complexity, from smallest to largest.
   1. $\log(\log(n))$
   2. $\log(n)$
   3. $\sum_{k=1}^{\log(n)} \frac{n^2}{2^k}$
   4. $n^{1/n}$

   Solution: Here is the right order:

   1. $n^{1/n}$, because $\lim_{n\to\infty} n^{\frac{1}{n}} = \lim_{n\to\infty} e^{\frac{\log n}{n}} = \exp\left(\lim_{n\to\infty} \frac{\log n}{n}\right) = 1$
   2. $\log(\log(n))$, because $\lim_{n\to\infty} \frac{\log\log n}{\log(n)}$ can be obtained using L'Hopital's rule as

   $$\lim_{n\to\infty} \frac{\log\log n}{\log(n)} = \lim_{n\to\infty} \frac{\frac{d}{dn}\log\log n}{\frac{d}{dn}\log(n)} = \lim_{n\to\infty} \frac{\frac{1}{n\log(n)}}{\frac{1}{n}} = 0$$

   3. $\log(n)$, as shown above.
   4. $\sum_{k=1}^{\log(n)} \frac{n^2}{2^k}$, because $\sum_{k=1}^{\log(n)} \frac{n^2}{2^k} \in [\frac{n^2}{2}, n^2]$.

2. Order the following functions in increasing order of asymptotic complexity, from smallest to largest.
   1. $n^{1.8}$
   2. n^2
   3. $\sum_{k=1}^{n} \frac{n^2}{k}$
   4. $3^{\ln(n)}$

   Solution: Here is the right order:

   1. $3^{\ln(n)}$
   2. $n^{1.8}$
   3. n^2
   4. $\sum_{k=1}^{n} \frac{n^2}{k}$

   because $3^{\ln(n)} = n^{\ln(3)}$, and $\ln(3) < 1.8$. Also, $\sum_{k=1}^{n} \frac{n^2}{k} \in \Theta(n^2\log(n))$.

3. Order the following functions in increasing order of asymptotic complexity, from smallest to largest.
   1. $n!$
   2. 2^n
   3. n^n
   4. $3^{\ln(n)}$

   Solution: Here is the right order:

   1. $3^{\ln(n)}$
   2. 2^n
   3. $n!$

4. n^n

because $n! \approx \sqrt{2\pi n}(n/e)^n$.

4. Consider the following function:

```
public int A(int n) {
      if (n == 1) return 1;
      else return A(n-1) * A(n-1);
}
```

Let T(n) be the run time performance for an input n. Which of the following is true?

a. $T(n) \in \Theta(2^n)$

b. $T(n) \in \Theta(n^2)$

c. $T(n) \in \Theta(n)$

d. $T(n) \in \Theta(\log(n))$

Solution: The right answer is a. Let $T(n)$ be the asymptotic time for the instance of size n. Then,

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

with initial condition $T(1) = 1$.

By induction, $T(2) = 3$, $T(3) = 7$. Claim that $T(n) = 2^n - 1$. Then,

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

which shows $T(n) \in \Theta(2^n)$.

5. Let $T(n)$ denote the asymptotic run time of this function.

```
int B(int n) {
     if (n == 0) return 1;
     if (B(n/2) >= 10)
          return B(n/2) + 10;
     else
          return 10;
}
```

As n increases, which of the following is true?

a. $T(n) \in \Theta(n)$

b. $T(n) \in \Theta(n^2)$

c. $T(n) \in \Theta(n\log(n))$

d. $T(n) \in \Theta(2^n)$

Solution: The right answer is a. Note that $B(0) = 1$, and $B(1) = 10$. For $n > 1$, we always have $B(n/2) >= 10$, so we have to follow the top branch of the if statement.

Thus, we have the recursion:

$$T(n) = 2 * T(n/2) + c$$

for some constant c. The 2 happens because we have to evaluate $B(n/2)$ once for the if condition, and another time for the return.

This fits the Master theorem model, we have that the solution is $T(n) \in \Theta(n)$.

6. Consider the function below. Let $T(n)$ denote the asymptotic run time of this function depending on n.

```
public int D(int n) {
     if (n == 0) return 1;
     if (n== 1) return 3;
     return D(n-1) + D(n-2)*D(n-2);
}
```

As n increases, which of the following is true?

a. $T(n) \in \Theta(n^2)$

b. $T(n) \in \Theta(n \log(n))$

c. $T(n) \in \Theta(n)$

d. $T(n) \in \Theta(2^n)$

Solution: The right answer is d. Note that $D(0) = 1$, and $D(1) = 3$. For $n > 1$,

A recursion for the asymptotic run time is

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

This is a constant coefficient difference equation, so we look for solutions of the form z^n for some roots z. Substituting into the equation, we find the roots must satisfy $z^2 - z - 2 = 0$, so the possible vales are $z = 2, z = -1$. The root with largest magnitude is 2, so $T(n) \in \Theta(2^n)$.

7. Consider the function below. Let $T(n)$ denote the asymptotic run time of this function depending on n.

```
public int E(int n) {
     if (n<= 1) return 1;
     int count = 3;
     int tmp = E(n/3);
     for (int k = 1; k < n; k++)
               if (tmp%k == 0)
                    count++;
     return E(n/3)*(count%2);
```

As n increases, which of the following is true?

a. $T(n) \in \Theta(n)$

b. $T(n) \in \Theta(n^2)$

c. $T(n) \in \Theta(n^{2/3})$

d. $T(n) \in \Theta(n^{\log_3(2)})$

Solution: The right answer is a. As before, let's derive a recursion: There are two calls to $E(n/3)$, and there is a loop of size n. Hence, the recursion is

$$T(n) = 2T(n/3) + cn$$

Using the Master's theorem, the answer is $\Theta(n)$.