CSE 321 – Introduction to Algorithm Design
Homework 02

1. **Thought of The Imitation Game**
   - The Imitation Game. This movie is one of my best ones because before everything, this movie is based on a true story, the story of Enigma and Alan Turing. In these time Enigma was thought as unbreakable to decoded. It was unbreakable because people do not believe that they can break it. There was a man, Alan Turing, who is think opposite. In the beginning of the movie, he says "Are you paying attention? Good. If you are not listening carefully, you will miss things. Important things. I will not pause, I will not repeat myself, and you will not interrupt me. You think that because you're sitting where you are, and I am sitting where I am, that you are in control of what is about to happen. You're mistaken. I am in control, because I know things that you do not know.". This tells lots of things but the significant one is the message that "Paying Attention" because everyone can listen but if you want to make difference you must do it carefully. By the way, he is very smart man but in the beginning, he was thought that he can solve everything himself but he was wrong. If you manage to be a team, you can do more important things than when you alone. There is a good proverb in Turkish that explain this "Bir elin nesi var iki elin sesi var." To sum up, this movie shows us lots of things but I think the most important ones are "Paying Attention" and "Being A Team".

2. **Solve following recurrence relations using master theorem.**
   - $x1(n)=0.5x1(\frac{n}{2}) + (\frac{1}{n})$ => a = 0.5, b = 2, d = -1. It cannot be solved with master theorem. Because the value of a is less than 1.
   - $x2(n)=3x2(\frac{n}{4}) + n\, logn$ => a = 3, b = 4, f(n) = $nlogn$. For this equation => f(n) is not polynomial so we must consider $n^{\log_b a}$ = $n^{\log_4 3}$ is less than f(n). So we must find a value of c that c < 1 from af($\frac{n}{b}$) <= cf(n). For c = $\frac{3}{4}$ that equation is fine. Because of that, for f(n) = Ω ($n^{\log_b a+\varepsilon}$) , the result is $x2(n) \in \Theta(f(n)) = \Theta(nlogn)$.
   - $x3(n)=3x3(\frac{n}{3}) + \frac{n}{2}$ => a = 3, b = 3, d = 1. According to these parameters the result is found with Master theorem like this: $a = b^d$ then $x3(n) \in \Theta(n^d logn)$. Thus, $x3(n) \in \Theta(nlogn)$.
   - $x4(n)= 6x4(\frac{n}{3}) + n^2 logn$ => a = 6, b = 3, f(n) = $n^2 logn$. For this equation => f(n) is not polynomial so we must consider $n^{\log_b a}$ = $n^{\log_3 6}$ = $n^{1+\log_3 2}$ is less than f(n). Because of that, for f(n) = Ω ($n^{\log_b a+\varepsilon}$) , the result is $x4(n) \in \Theta(f(n)) = \Theta(n^2 logn)$.
   - $x5(n)= 4x5(\frac{n}{2})+\frac{n}{logn}$ => a = 4, b = 2, f(n) = $\frac{n}{logn}$. For this equation => f(n) is not polynomial so we must consider $n^{\log_b a}$ = $n^{\log_2 4}$ = n^2 is bigger than f(n). Because of that, the result is $x5(n) \in \Theta(n^{log_b a}) = \Theta(n^{log_2 4}) = \Theta(n^2)$.

- $x6(n) = 2^n x6(\frac{n}{2}) + n^n \Rightarrow$ a = 2^n, b = 2, d = n. According to these parameters the result cannot found with Master theorem because a is not constant and we cannot verify that a is bigger than 1.

3. **Consider the following algorithm implemented in python:**

```
def chocolateAlgorithm(n):
    #Input is a positive integer n
    if n==1:
        return 1
    else:
        return chocolateAlgorithm(n-1) + 2 * n -1
```

a) **Set up a recurrence relation for this function's values and solve it to determine what this algorithm computes**
   i. The recurrence relation for this algorithm is $T(n) = T(n-1) + 2 * n - 1$, $T(1) = 1$ and $n > 1$
   ii. $T(2) = T(1) + 2 * 2 - 1 = 4$
   iii. $T(3) = T(2) + 2 * 3 - 1 = 9$
   iv. $T(4) = T(3) + 2 * 4 - 1 = 16$
   v. According to these results, $T(n) = n^2$. That's mean this compute the square for given number.

b) **Set up a recurrence relation for the number of multiplications made by this algorithm and solve it.**
   i. Let's assume that $T_1(n)$ for number of multiplications.
   ii. Then the recurrence relation is $T_1(n) = T_1(n-1) + 1$, $T_1(1) = 0$, $n > 1$
   iii. $T_1(2) = T_1(1) + 1 = 1$
   iv. $T_1(3) = T_1(2) + 1 = 2$
   v. $T_1(4) = T_1(3) + 1 = 3$
   vi. $T_1(5) = T_1(4) + 1 = 4$
   vii. According to these results, $T_1(n) = n - 1$.

c) **Set up recurrence relation for the number of additions/subtractions made by this algorithm and solve it.**
   i. Let's assume that $T_2(n)$ for number of additions/subtractions.
   ii. Then the recurrence relation is $T_2(n) = T_2(n-1) + 2$, $T_1(1) = 0$, $n > 1$
   iii. $T_2(2) = T_2(1) + 2 = 2$
   iv. $T_2(3) = T_2(2) + 2 = 4$
   v. $T_2(4) = T_2(3) + 2 = 6$
   vi. $T_2(5) = T_2(4) + 2 = 8$
   vii. According to these results, $T_2(n) = 2 * n - 2$

4. **The code file for Towers of Hanoi is uploaded.**

5. **Consider the problem of finding rotten walnut.**
   - The code file for finding rotten walnut is uploaded.
   - B(n) and W(n):
     i. The best case (B(n)) of the algorithm is $B(n) = 1 \in \Theta(1)$ if the rotten walnut is the middle element.
     ii. To analyse the worst case of the algorithm firstly we need to suppose that $n = 2^k - 1$. If rotten walnut is not in the list or rotten walnut = L [0] or rotten walnut = L [n - 1] then the worst case occurs. Thus, the worst case complexity W(n) $\in \Theta(logn)$ for all n.

6. **Solve the following recurrence relations**
   a) Using backward/forward substitution:
      i. $\boldsymbol{T1(n)=3T1(n-1)}$ $\quad$ $\boldsymbol{for\ n>1, T1(1) = 4}$
      - $T1(2)$ =3*T1(1) = 12 = 3.4
      - T1(3) = 3*T1(2) = 36=3^2 *4
      - T1(4) = 3*T1(3) = 3^3 *4
      - T1(5) = 3 * T1(4) = 3^4 *4
      - According to these results, <u>T1(n) = $3^{n-1} * 4$</u>
      - Then control for T1(n) = <u>$3^{n-1} * 4$</u>, this equation T1(n) = $3T1(n-1)$ is correct.
      - They are equivalent so T1(n) = $3^{n-1} * 4$.

      ii. $\boldsymbol{T2(n)=T2(n-1) + n}$ $\quad$ $\boldsymbol{for\ n>1, T2(0) = 0}$
      - T2(1) = T2(0) + 1 = 1
      - T2(2) = T2(1) + 2 = 3
      - T2(3) = T2(2) + 3 = 6
      - T2(4) = T2(3) + 4= 10....
      - According to these results, T2(n) = T2(1) + $\sum_{i=2}^{n} i = n * \frac{n+1}{2}$
      - Then control for T2(n) = $n * \frac{n+1}{2}$, this equation T2(n) = $T2(n-1)$ + n is correct.
      - They are equivalent so T2(n) =$n * \frac{n+1}{2}$.

      iii. $\boldsymbol{T3(n)= T3(\frac{n}{2}) + n}$ $\quad$ $\boldsymbol{for\ n>1, T3(1) = 0\ (solve\ for\ n=2^k)}$
      - $T3(2^k) = T3(2^{k-1}) + 2^k$
      - $T3(2^k) = T3(2^{k-2}) + 2^{k-1} + 2^k$
      - $T3(2^k) = T3(2^{k-3}) + 2^{k-2} + 2^{k-1} + 2^k$
      - $T3(2^k) = T3(2^{k-m}) + 2^{k-(m-1)} + 2^{k-(m-2)} + \dots \dots + 2^{k-1} + 2^k \quad m > 0$
      - $T3(2^k) = T3(2^{k-k}) + 2^{k-(k-1)} + 2^{k-(k-2)} + \dots \dots + 2^{k-1} + 2^k$
      - $T3(2^k) = T3(2^0) + 2^1 + 2^2 + \dots \dots + 2^{k-1} + 2^k$
      - $T3(2^k) = \sum_{i=0}^{k-1} 2^i + 2^k - 1 = 2^k - 1 + 2^k - 1$
      - $T3(2^k) = 2 * 2^k - 2$ => $T3(n) = 2 * n - 2$

b) Using the properties of linear homogeneous/inhomogeneous equations:

i. $T_1(n) = 6T_1(n\text{-}1) - 9T_1(n\text{-}2)$, $T_1(0) = 1$, $T_1(1) = 6$

- According to this linear homogeneous relation, we can say that $T_1(n) = x^2$, $T_1(n - 1) = x$, $T_1(n - 2) = 1$.
- Then we have this equation $x^2 = 6x - 9$. The roots of this equation are $x1 = 3$, $x2 = 3$.
- Then we can say that, $T_1(n) = \alpha 3^n + \beta n 3^n$ by rules.
- $T_1(0) = 1 \Rightarrow$ $\underline{1 = \alpha}$ and $T_1(1) = 6 \Rightarrow$ $\underline{1 = \beta.}$
- Then the equation is $T_1(n) = 3^n + n3^n$.

ii. $T_2\,(n) = 5T_2\,(n-1) - 6T_2\,(n-2) + 7^n$

- For this question, we have non-homogeneous equation. Because of that $T_2\,(n) = T_2^p(n) + T_2^h(n)$.
- To find homogeneous part, we must do same steps (above question). $T_2(n) = x^2$, $T_2(n - 1) = x$, $T_2(n - 2) = 1$
- Then we have this equation $x^2 = 5x - 6$. The roots of this equation are $x1 = 3$, $x2 = 2$.
- Then we can say that, $T_2^h(n) = \alpha 3^n + \beta 2^n$ by rules.
- To find non-homogeneous part, let say that $T_2^p(n) = X * 7^n$. Then find the value of X.
- $X7^{n+2} - 5X7^{n+1} + 6X7^n = 7^n$ from this equation X = $\frac{1}{20}$ and $T_2^p(n) = \frac{1}{20} * 7^n$
- Then the equation is $T_2\,(n) = \alpha 3^n + \beta 2^n + \frac{1}{20} * 7^n$.