You are allowed to discuss with others but not allow to use references other than the course notes and reference books. Please list your collaborators for each questions. Write your own solutions and make sure you understand them.

There are 60 marks in total (including the bonus). The full mark of this homework is 50. You submission should be in PDF format generated by . We suggest to use the template available at:

https://www.overleaf.com/read/tsxxqdgjzhxx

Enjoy :).

Problem 1: Asymptotics [5 marks]

Arrange the following functions in order of increasing growth rate, with $g(n)$ following $f(n)$ in your list if and only if $f(n) = O(g(n))$.

- $\log_2 \log_2 n$
- n^3
- $n^{\log_2 n}$
- $\sqrt{\log_2 n}$
- $n^{1/\log_2 n}$

Problem 2: Solving recurrences [15 marks]

(a) (10 marks) Find an asymptotically tight bound of the following recurrence relations. Justify your answers by naming a particular case of the Master method, or by iterating the recurrence, or by using the substitution method. Assume that the base cases can be solved in constant time.

(i) $T(n) = 2T(n/4) + 2n$
(ii) $T(n) = T(n-2) + n^2$
(iii) $T(n) = 2T(2n/3) + T(n/3) + n^2$

(b) (5 marks) Consider the recurrence relation $C_0 = 0$ and

$$C_n = n + 1 + \frac{2}{n} \sum_{k=0}^{n-1} C_k.$$

Find an explicit formula for C_n.

Problem 3: Fibonacci-3 [10 marks]

Consider the recurrence relation $F_{n+3} = F_{n+2} + F_{n+1} + F_n$, with the initial state $F_0 = 0, F_1 = 0, F_2 = 1$.

(a) Prove that

$$\begin{pmatrix} F_n \\ F_{n+1} \\ F_{n+2} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}^n \cdot \begin{pmatrix} F_0 \\ F_1 \\ F_2 \end{pmatrix}.$$

(b) So, in order to compute F_n, it suffices to raise this 3×3 matrix, called X, to the nth power. Show that $O(\log n)$ matrix multiplications suffice for computing X^n.

Problem 4: Recurrences in programs [10 marks]

Consider the following two programs:

```
int F(int x) {
  assert(x>=1);

  if (x == 1 || x == 2)
    return 1;
  else
    return 2*F(x-1) - F(x-2);
}

void Hanoi(int disk, int source, int dest, int spare) {
  if (disk == 1) {
    return;
  }
  else {
    Hanoi(disk - 1, source, spare, dest);
    Hanoi(disk - 1, spare, dest, source);
  }
}
```

(a) How many times is the function `F` is called when invoking `F(n)` with $n \geq 1$?

(b) How many times is the function `Hanoi` called when invoking `Hanoi(n,0,0,0)` with $n \geq 1$?

Problem 5: Summations [10 marks + 10 marks]

(a) (10 marks) Let us be given three sequences of integers, say A, B, and C, each of length n. Devise an algorithm to find whether there are three numbers $a \in A$ and $b \in B$ and $c \in C$ such that $a + b + c = 0$. You will get full marks if the algorithm is of $O(n^2)$ complexity and it is proven to be correct.

(b) **(Bonus: 10 marks)** Let us be given four sequences of integers, say A, B, C, and D, each of length n. Devise an algorithm to find whether there are four numbers $a \in A$ and $b \in B$ and $c \in C$ and $d \in D$ such that $a + b + c + d = 0$. You will get full marks if the algorithm is of $O(n^2 \log n)$ complexity and it is proven to be correct.