COMP2119 Introduction to Data Structures and Algorithms
Practice Problem Set 1 - Recursion, Mathematical Induction & Algorithm Analysis

Please do NOT submit the answer of this part.

1. For each of the following recurrence relations, solve the recurrence and state the Θ bound.

   (a) $T(n) = T(n-2) + n$; $T(0) = T(1) = 0$

   (b) $T(n) = T(\sqrt{n}) + 2$; $T(1) = T(2) = 0$ (You may assume that $n = 2^{2^k}$ where k is a power of 2.)

   (c) $T(n) = 5T(n/5) + n/5$; $T(1) = 0$ (You may assume that n is a power of 5.)

   (d) $T(n) = T(n-1) + \log_2 n$; $T(1) = 0$

   (e) $T(n) = T(\frac{2n}{5}) + T(\frac{3n}{5}) + n$; $T(n) = 0$ if $n \leq 1$

2. Use mathematical induction to show that $n! > 2^n$ for all $n \geq 4$, where n is a positive integer.

3. In lecture, the sequence of Fibonacci numbers is discussed, i.e. $P(1) = P(2) = 1, P(n) = P(n-1) + P(n-2)$ for all $n \geq 3$. Use mathematical induction to prove that for all positive integers,

$$P(n) = \frac{1}{\sqrt{5}}[(\frac{1+\sqrt{5}}{2})^n - (\frac{1-\sqrt{5}}{2})^n]$$

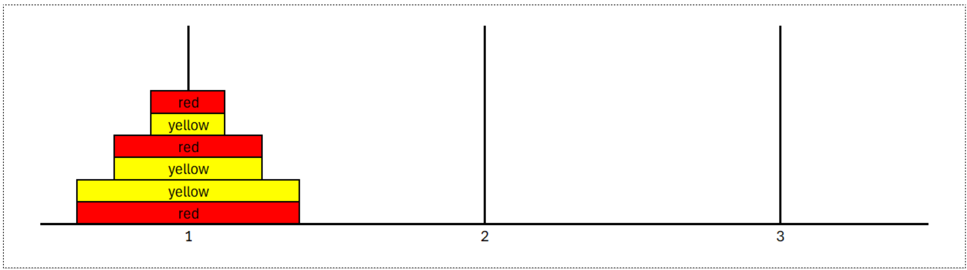
4. The following pseudocode calculates the greatest common divisor of two positive integers `a` and `b`. Use M.I. to prove the correctness the gcd function.

```
gcd(a, b):
    if b == 0:
        return a
    else:
        return gcd(b, a mod b)
```

5. In lecture, the Towers of Hanoi problem is discussed. Consider the following modified version of the problem:

There are n types of disk sizes and 3 pegs. For each of the disk size, there are one red disk and one yellow disk. All the disks are placed in order of size on the same peg. You are required to transfer the disks from the original peg to the destination peg in as few moves as possible, while keeping the original sequence of the disks. The constraints on moving the disks are the same as the original Towers of Hanoi problem mentioned in the lecture notes, i.e., no disk may be placed on top a disk that is smaller than it. For two disks of the same size, it is always safe to place one on top of the other regardless of their colours.

Figure 1 shows the initial configuration and target configuration of an instance when $n = 3$.



(a) Initial Configuration

(b) Target Configuration

Figure 1: Initial Configuration and Target Configuration of an Instance when $n = 3$.

Let $P(n)$ be the number of moves required to move n types of disks for the problem above.

(a) Find $P(1)$, $P(2)$.

(b) Describe, in words, a recursive algorithm to solve the above-mentioned version of Towers of Hanoi.

(c) Hence, form a recurrence relation regarding to $P(n)$.

(d) Solve the recurrence relation obtained in part (c).

(e) Hence, give an upper bound for the time-complexity of the algorithm in part (b) in terms of big-O notation.

Further Practice from the Textbook:

To gain more practice, here are a list of supplementary problems you may attempt from the course textbook **Introduction to Algorithms, 3rd edition**:

1. Solving Recurrence Relations:
   Exercise 2.3-3, 4.3-1, 4.3-2, 4.3-3, 4.3-6, 4-1, 4.3, 7.2-1, 7.4-1

2. Asymptotic Bounds:
   Exercise 3.1-1, 3.1-4, 3.2-5, 3.2-8, 3-2, 3-3, 3-4

3. Recursive Algorithm Design:
   Exercise 2.3-7, 9.3-8

However, due to copyright issues, we cannot directly upload the textbook or solution here. You may search the answers of these exercises online (E.g. By searching "Introduction to algorithms 3rd edition solution" on google).