ensia The National School of Artificial Intelligence المدرسة الوطنية العليا للذكاء الاصطناعي

2nd Year /Semester3 (2023-2024)

Data Structures & Algorithms 2
Tutorial 2
Algorithm Analysis

OBJECTIVES

- Compute the computational complexity for an algorithm

Exercise 1

Are the following formulas True or False, Justify your answer :

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

Exercise 2

Order the following functions by growth rate:

N, $\sqrt{N}$, $N^{1.5}$, N^2, N log N, N log log N , N $\log^2$N , N log(N^2), 2/N , 2N , $2^{N/2}$,

37 , N^2log N , N^3 .

Indicate which functions grow at the same rate. Give better insights of how you solved this?

Exercise 3

Suppose $T_1(N) = O(f(N))$ and $T_2(N) = O(f(N))$. Which of the following are true?

1. $T_1(N) + T_2(N) = O(f(N))$
2. $T_1(N) - T_2(N) = o(f(N))$
3. $\frac{T_1(N)}{T_2(N)} = O(1)$
4. $T_1(N)= O(T_2(N))$

Exercise 4

In a recent court case, a judge cited a city for contempt and ordered a fine of $2 for the first day. Each subsequent day, until the city followed the judge's order, the fine was squared (i.e., the fine progressed as follows: $2, $4, $16, $256, $65 536, . . .).

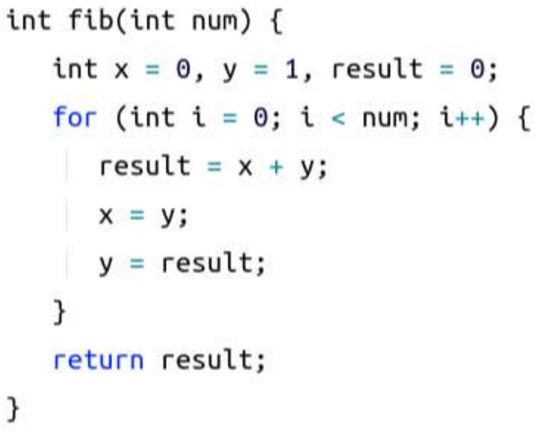
1. What would be the fine on day N?

2. How many days would it take for the fine to reach D dollars (a Big-Oh answer will do)?

Exercise 5

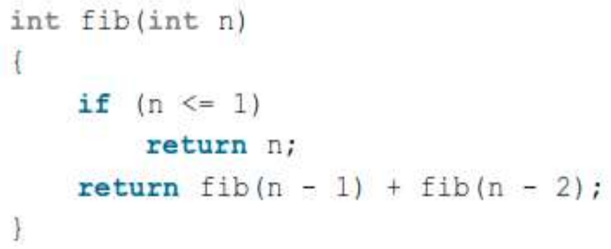
Estimate the complexity big O for the following functions:

1)- Iterative Fibonacci Function V1



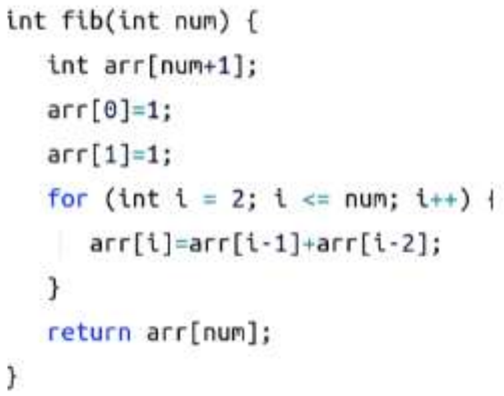
```
int fib(int num) {
   int x = 0, y = 1, result = 0;
   for (int i = 0; i < num; i++) {
      result = x + y;
      x = y;
      y = result;
   }
   return result;
}
```

2)- Recursive Fibonacci Function



```
int fib(int n)
{
    if (n <= 1)
        return n;
    return fib(n - 1) + fib(n - 2);
}
```

3)-Iterative Fibonacci Function V2



```
int fib(int num) {
   int arr[num+1];
   arr[0]=1;
   arr[1]=1;
   for (int i = 2; i <= num; i++) {
      arr[i]=arr[i-1]+arr[i-2];
   }
   return arr[num];
}
```

4)-Fibonacci Function using Exponentiation:

If we compute it mathematically using the following formula:

$$F_n = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right)$$

Exercise 6

- Write the fast exponentiation routine without recursion using the squaring method.
- Estimate the complexity of the algorithm.