Practical Data Structures and Algorithms

Supplement A

Instructor: Chien-Yu Chen

BIME@NTU

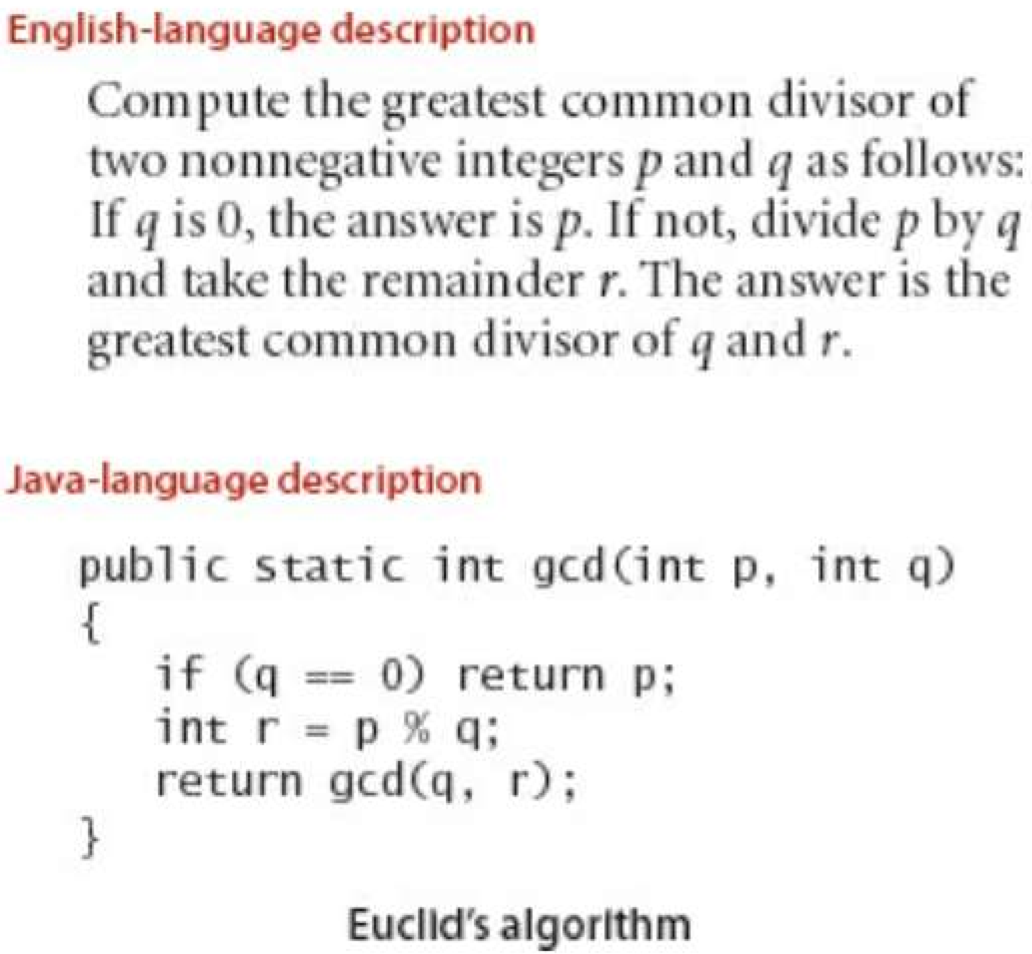
Monday 14:20-17:10

Recursion

Recursive function

Chien-Yu Chen, BIME@NTU

Finding the greatest common divisor of two numbers



English-language description

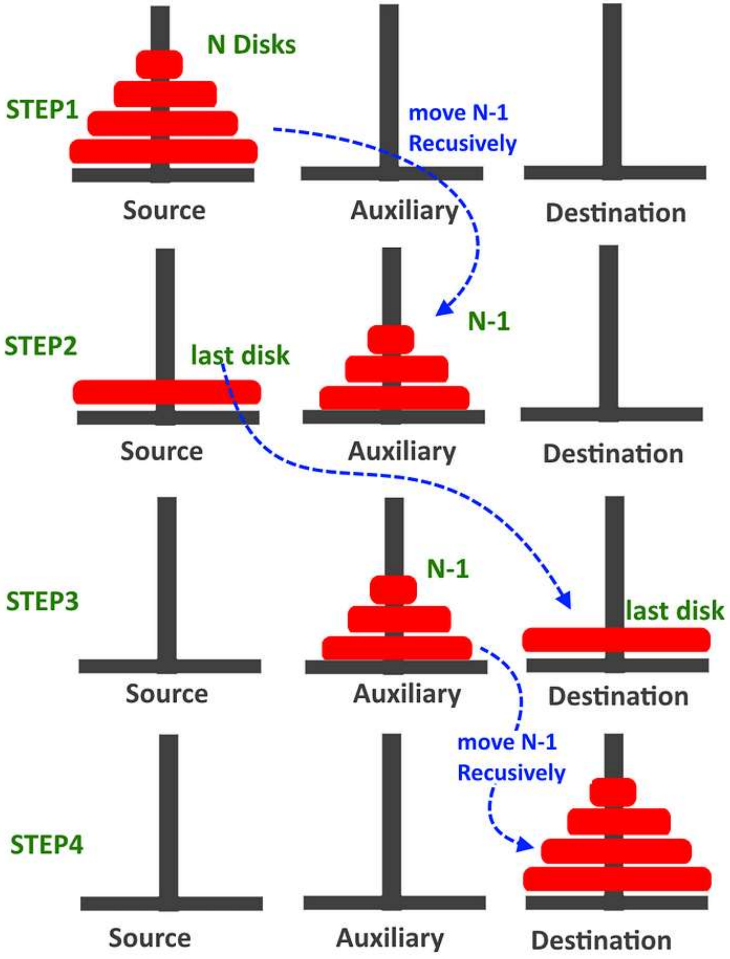
Compute the greatest common divisor of two nonnegative integers *p* and *q* as follows: If *q* is 0, the answer is *p*. If not, divide *p* by *q* and take the remainder *r*. The answer is the greatest common divisor of *q* and *r*.

Java-language description

```
public static int gcd(int p, int q)
{
   if (q == 0) return p;
   int r = p % q;
   return gcd(q, r);
}
```

Euclid's algorithm

Tower Of Hanoi



https://medium.com/@jamalmaria111/tower-of-hanoi-js-algorithm-3f667fa46f0f

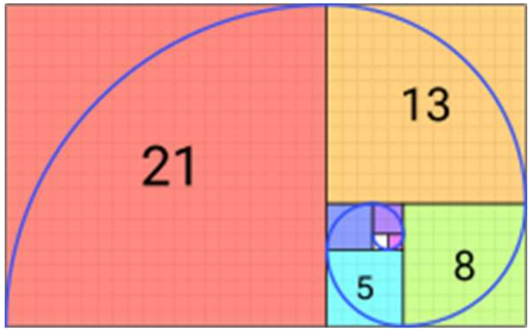
The Fibonacci numbers may be defined by

$$F_0 = 0, \quad F_1 = 1,$$

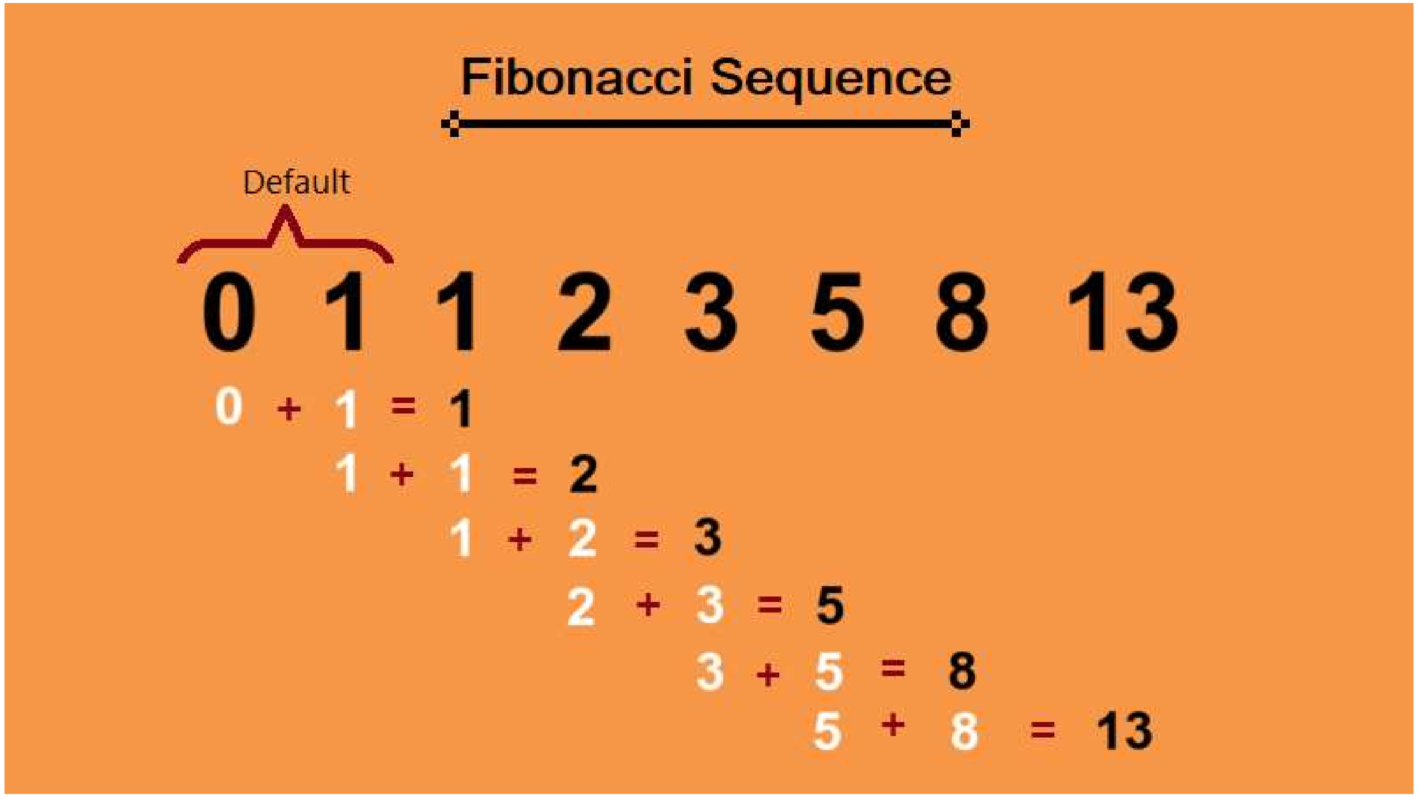
and

$$F_n = F_{n-1} + F_{n-2}$$

for $n > 1$.



https://en.wikipedia.org/wiki/Fibonacci_sequence



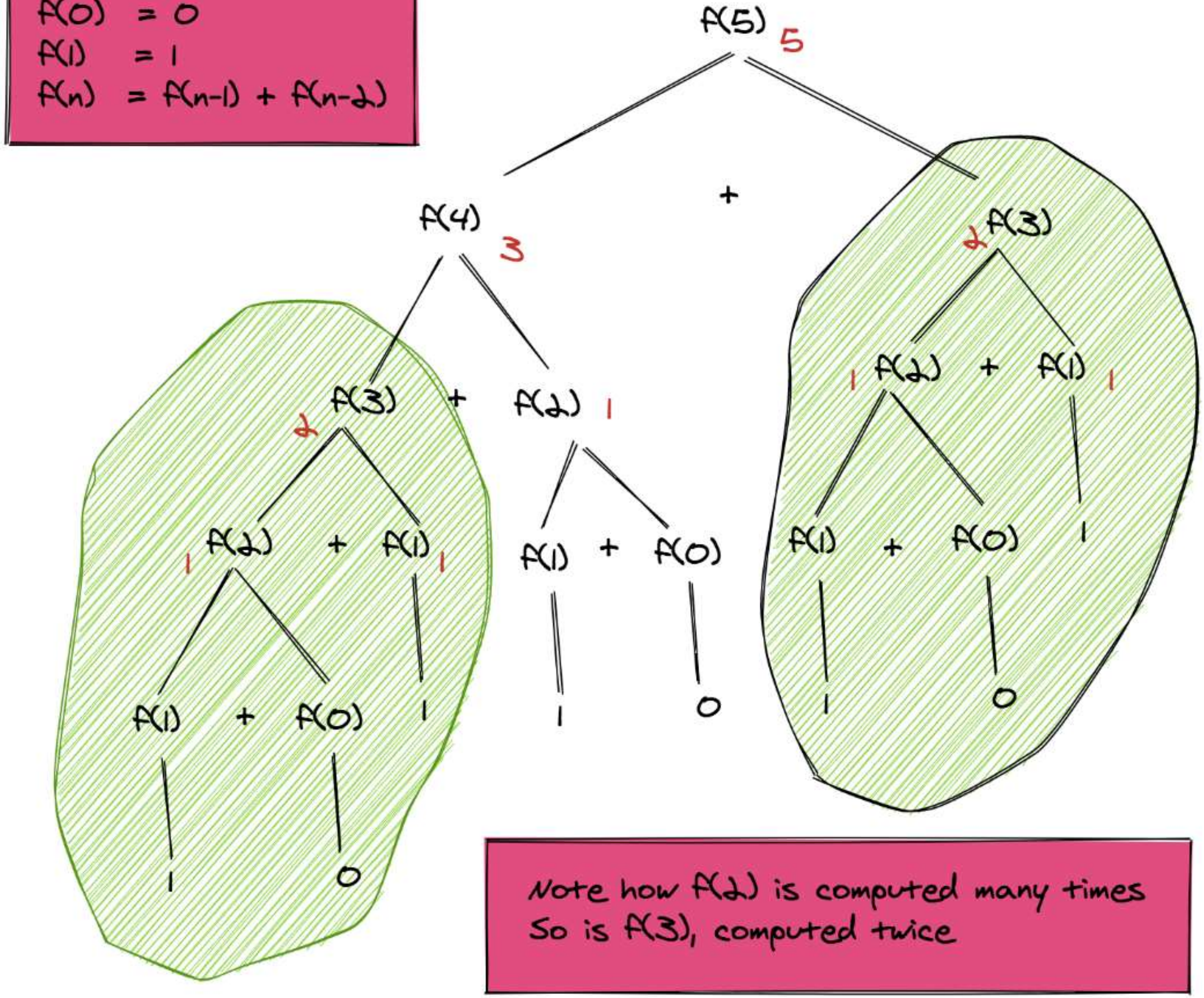
https://tecadmin.net/what-is-fibonacci-sequence/

Asking ChatGPT to write a recursive function to calculate Fibonacci numbers

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

Fibonacci Numbers

f(0) = 0
f(1) = 1
f(n) = f(n-1) + f(n-2)



Note how f(2) is computed many times
So is f(3), computed twice

https://algodaily.com/challenges/fibonacci-sequence

Dynamic Programming (DP)

DP refers to simplifying a complicated problem by breaking it down into simpler sub-problems in a recursive manner.

Chien-Yu Chen, BIME@NTU

Two key attributes of DP

- There are two key attributes that a problem must have in order for dynamic programming to be applicable: optimal substructure and overlapping sub-problems.
  - If a problem can be solved by combining optimal solutions to non-overlapping sub-problems, the strategy is called "divide and conquer" instead.
  - This is why merge sort and quick sort are not classified as dynamic programming problems.

Dynamic programming - Wikipedia

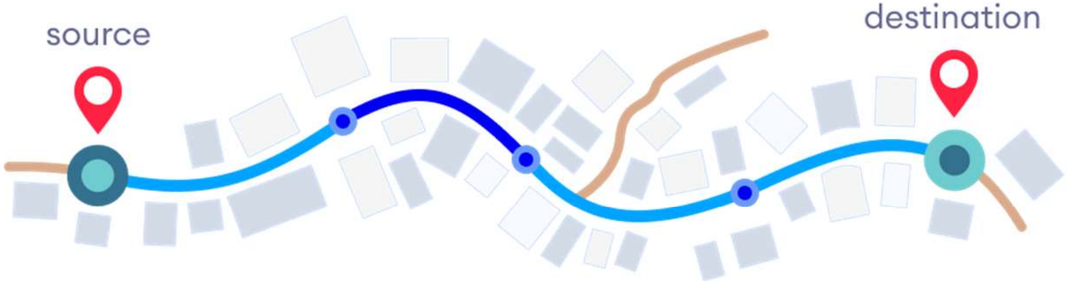
Asking ChatGPT to calculate Fibonacci numbers by DP

```
n = 2000;
int[] fSeries = new int[n];
fSeries[0] = 0;
fSeries[1] = 1;
for (int i = 2; i <= n; i++) {
    fSeries[i] = fSeries[i - 1] + fSeries[i - 2];
}
```

Asking ChatGPT to write a recursive function with memorization to calculate Fibonacci numbers

```
public static int fibonacci(int n) {
    if (memo[n] != 0) {
        return memo[n];
    }
    if (n <= 1) {
        memo[n] = n;
    } else {
        memo[n] = fibonacci(n - 1) + fibonacci(n - 2);
    }
    return memo[n];
}
```

Finding shortest path



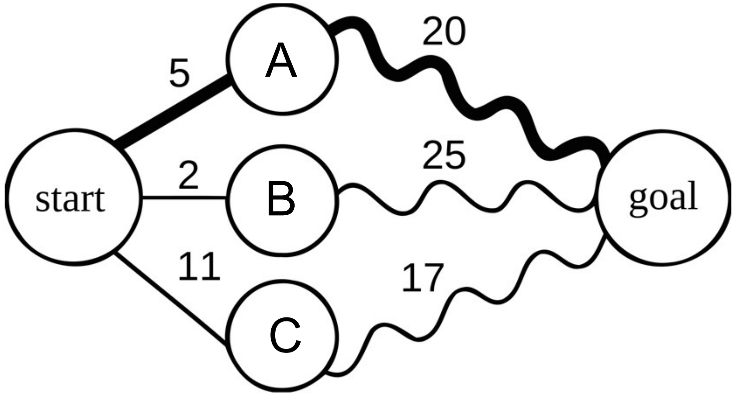
🟦 the shortest path between the source and destination

🟦 a subpath which is also the shortest path between its source and destination

https://www.programiz.com/dsa/dijkstra-algorithm

Identifying sub-problems

- Defining a function:
  - ShortestDistanceToGoal(start)
    - ShortestDistanceToGoal(A)
    - ShortestDistanceToGoal(B)
    - ShortestDistanceToGoal(C)


start
A
B
C
goal
5
2
11
20
25
17