Intro to dynamic Programming

CPMSOC

Yiheng You, Bharat Singla

Arc CLUBS SUPPORTED BY Arc INDEPENDENTLY RUN

Attendance

CPMSOC

Arc CLUBS
SUPPORTED BY Arc
INDEPENDENTLY RUN

SCAN ME

MOTIVATING PROBLEM

The fibonacci numbers are defined fib(0)=1, fib(1)=1, and fib(n)=fib(n-1)+fib(n-2).

e.g. 1,1,2,3,5,8,11,...

Given x, output the x-th fibonacci number.

Recursion?

Implement the recursive formula straight-forward.

CPMSOC

ARC CLUBS
SUPPORTED BY Arc
INDEPENDENTLY RUN

Recursion?

Implement the recursive formula straight-forward.

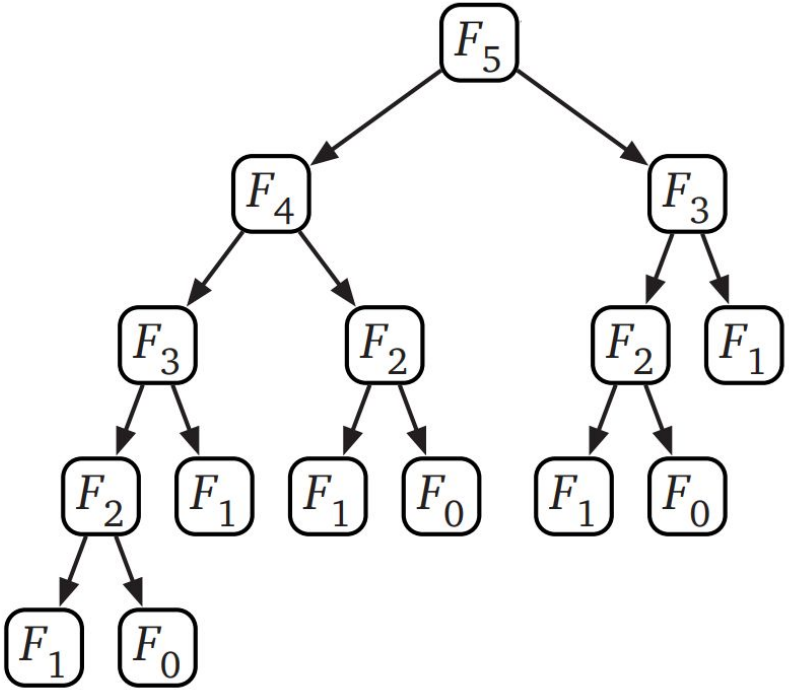
```
int fib(int n) {
    if (n <= 1) return 1;
    return fib(n - 1) + fib(n - 2);
}
```

So what's the issue?

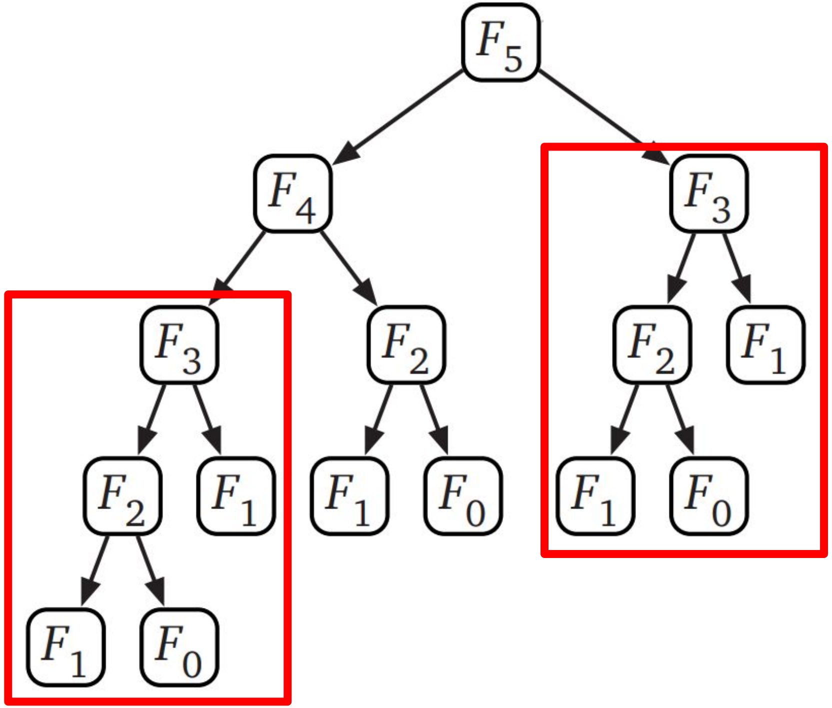
Try to run for a number x > 30.

The program can't finish running…

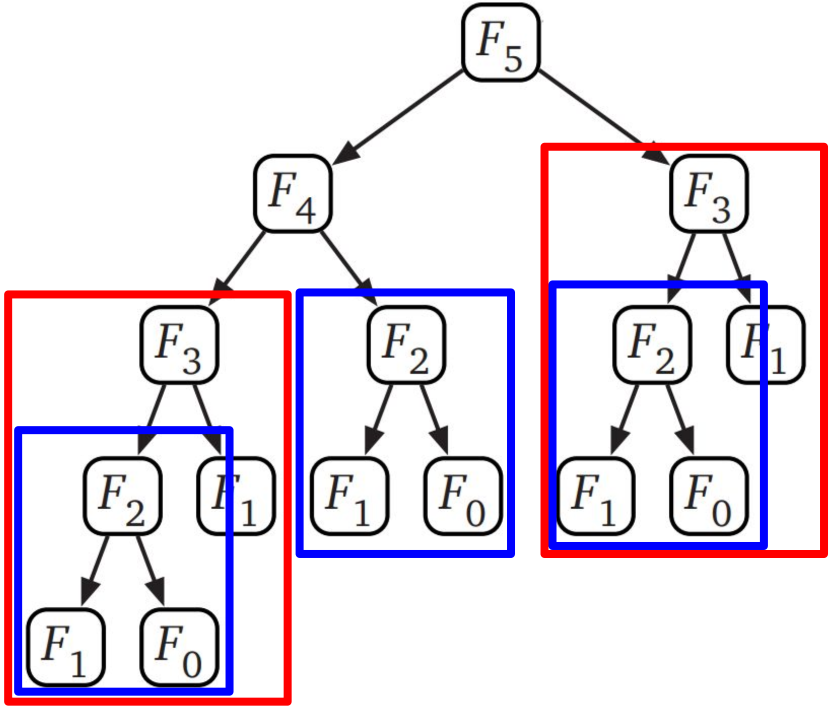
Let's see what's happening



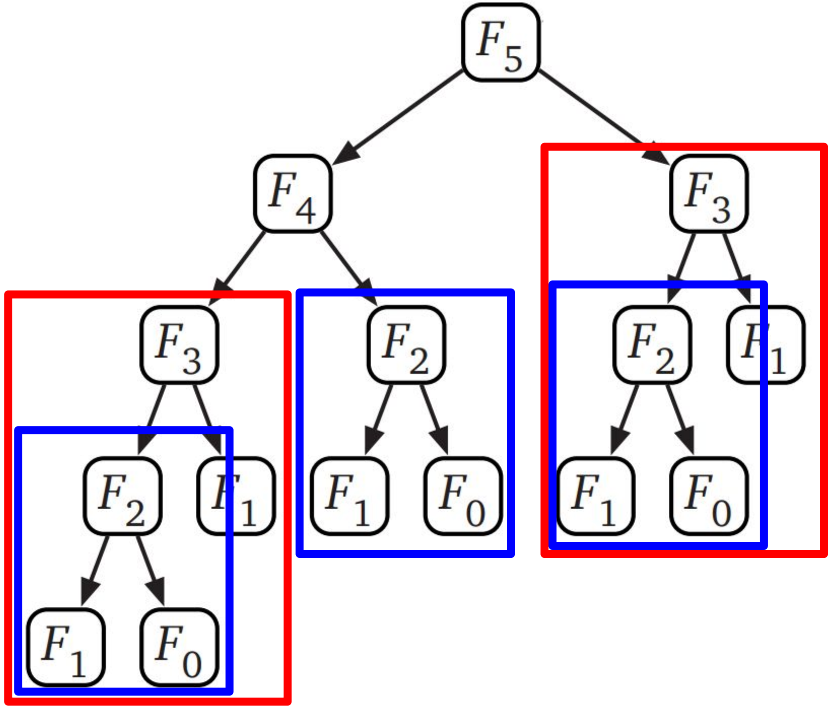
Let's see what's happening



LET'S SEE WHAT'S HAPPENING



Let's see what's happening



Worst case we're calculating 2^x states!

O(2^x)

MEMOISATION!

Let's not recalculate the values we've already calculated before.

MEMOISATION!

Let’s not recalculate the values we’ve already calculated before.

```
int dp[MXX];
bool seen[MXX];
int fib(int n) {
    if (n <= 1) return 1;
    if (seen[n]) return dp[n];
    dp[n] = fib(n - 1) + fib(n - 2);
    return dp[n];
}
```

CPMSOC

Arc CLUBS SUPPORTED BY Arc INDEPENDENTLY RUN

Why is this better?

We only calculate each state from 0~x once. O(x)

This new method will now easily pass for much larger values of x. Yay!

What insight does this give to DP?

- Similar sub-problems
- Similar sub-structures

We break the original problem up into smaller, manageable problems that we combine!

Coin Change problem

Given some denomination of coins, and a value, what is the minimum number of coins to make that value?

Example:

Value: 12

Denominations: 1, 2, 5

Idea

Let’s always take the largest denomination coin that we can take, then move on to the next.

Example:

Value: 12

Denominations: 1, 2, 5

Solution: 3 coins (5, 5, 2)

Does this always work?

CPMSOC

Arc CLUBS SUPPORTED BY Arc INDEPENDENTLY RUN

No

Consider:

Value: 11

Denominations: 1, 5, 7

No

Consider:

Value: 11

Denominations: 1, 5, 7

If we took greedily from biggest denomination: 5 coins (7, 1, 1, 1, 1)

Optimal solution: 3 coins (5, 5, 1)

Any Other ideas?

Let's think with DP

If we can't solve the problem for value V, let's solve it for value V-c[i], where c[i] is the denomination for each coin.

A recursive relation is being generated here!

Let's think with DP

If we can't solve the problem for value V, let's solve it for value V-c[i], where c[i] is the denomination for each coin.

A recursive relation is being generated here!

dp[V] = min{dp[V - c[i]] + 1}

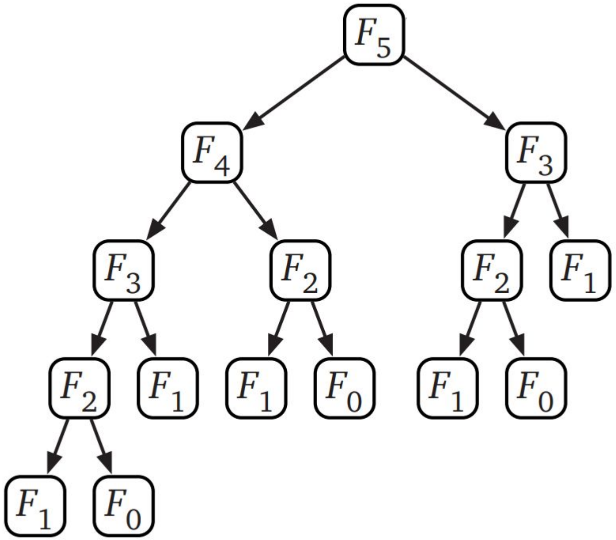
When do we stop?

```
dp[0] = 0
```

```
It requires 0 coins to form a total value of 0!
```

Top Down DP

We recursively call “states” that may have not been calculated yet.



Botton Up Dp

We build our dp calculations upwards with the states that have already been calculated.

Let’s see this in action with the Coin Change Problem!

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	INF	INF	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	INF	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	INF	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	INF	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	1	INF	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	1	2	INF	INF	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	1	2	3	INF	INF

Coin Change Problem Visualisation

CPMSOC

Arc CLUBS SUPPORTED BY Arc INDEPENDENTLY RUN

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	1	2	3	2	INF

Coin Change Problem Visualisation

Denominations: 1, 5, 7

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	1	2	1	2	3	2	3

Thanks for attending!

Attendance ------>

SCAN ME