COL 100 - L8

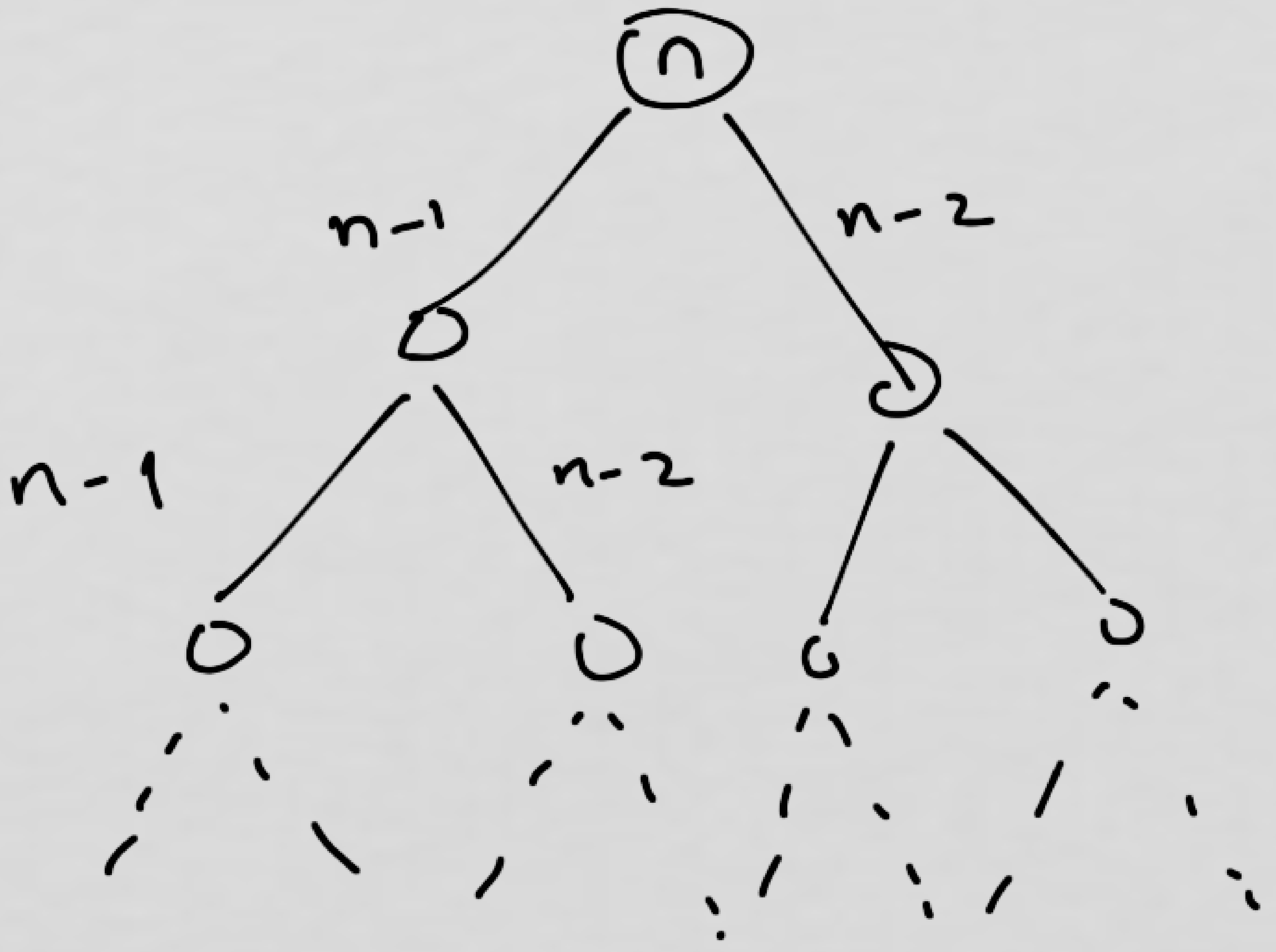
Let us revisit coin exchange problem

$$cc(a, n) = \begin{cases} 0 & \text{if } n = 0 \wedge a > 0 \\ 1 & \text{Else if } a = 0 \wedge n > 0 \\ cc(a-d, n) + cc(a, n-1) & \text{Else} \end{cases}$$

```
fun cc (a, n) =
        if (a=0) then 1
        else if (n=0 orelse a<0) then 0
             else
                 cc (a, n-1) +
                 cc (a- denomination (n), n)

fun denomination (n) =
        if (n = 1) then 1
        else if (n= 2) then 5
             else if (n= 3) then 10
                       :
```

Time Complexity



$$\left.\begin{array}{l} 2^0 \\ 2^1 \\ \vdots \\ 2^n \end{array}\right\}$$ max height of depth n

$O(2^n)$

so in the coin change problem

$$O(2^{a+n})$$

I change the problem slightly:

minimum number of coins to make change

i.e.

- I have an amount 'a'
- I have n type of coins with different denominations $v_1, v_2, \ldots, v_n$
- Assume $v_1 = 1$ [always make change for any amount]
- find minimum number of coins to make change for a.

Eg: Denomination : 1, 5, 10, 25, 50

Amount : 42

Min : 5 coins (25, 10, 5, 1, 1)

Developing a recursive solution

- Let cc (a) be the function that will make change for a with minimum number of coins.
- Idea: divide the problem into subproblems.

How to subdivide?

Let

$$cc(a) = K$$

$$\underbrace{\circ \quad \circ \quad \cdots \quad \bullet}_{K \text{ coins}}$$

the value of the last coin can be

v_1 or v_2 or ..... or v_n

— Indeed we don't know which of these denominations is the correct answer

What do we Know?

But we do know that

- if the last coin is v_1 then the remaining coins will be smallest number of coins to make the change for $a - v_1$
- if the last coin is v_2 then the remaining coins will be smallest number of coins to make the change for $a - v_2$

⋮ and so on!

Original problem

$\min cc(a, n)$

$\min cc(a - v_1, n)$

$\min cc(a - v_2, n)$

$\vdots$

$\min cc(a - v_n, n)$

Be careful

- We can use v_i only if

  $a \geq v_i$

- if $(a \geq v_1)$ then val $sol_1 = cc(a - v_1, n) + \underline{\underline{1}}$

Used one coin in each step

Is that all?

- final solution

$$\min(sol_1, sol_2, \ldots, sol_n)$$

H.W: SML Code & Time Complexity?

McCarthy 91 function

$$M(n) = \begin{cases} n - 10 & \text{if } n > 100 \\ \underbrace{M(M}_{\text{Nested Recursion}}(n+11)) & \text{if } n \leq 100 \end{cases}$$

Nested Recursion

$$\forall n \leq 100 \quad M(n) = 91$$

Can we prove it?

- Apply strong induction

$$90 \le n \le 100$$

$$M(n) = M(M(n+11))$$

$$= M(n+11-10) \quad \because n+11 > 100$$

$$= M(n+1)$$

$$\Rightarrow \underbrace{M(n) = M(101)}_{} = 91$$

↳ Use this as the base case!

For the I.S.

let $n \leq 89$,

assume $M(i) = 91 \quad \forall \; n < i \leq 100$

then

$$
\begin{aligned}
M(n) &= M(M(n+11)) && [\text{by def}^n] \\
&= M(91) && [\text{by hypothesis}] \\
&= 91 && [\text{by the base case}]
\end{aligned}
$$

Tail Recursive variant of McCarthy 91

$$M_{it}(n) = M\_iter(n, 1)$$

where

$$M\_iter(n, c) = \begin{cases} n & \text{if } c = 0 \\ M\_iter(n-10,\ c-1) & \text{if } n > 100 \text{ and } c \neq 0 \\ M\_iter(n+11,\ c+1) & \text{if } n \leq 100 \text{ and } c \neq 0 \end{cases}$$