Q You go to a party of N people

2 rules:

1. Either enjoy the party alone.

2. ______ as a pair.

Find the no. of ways of N people to party!

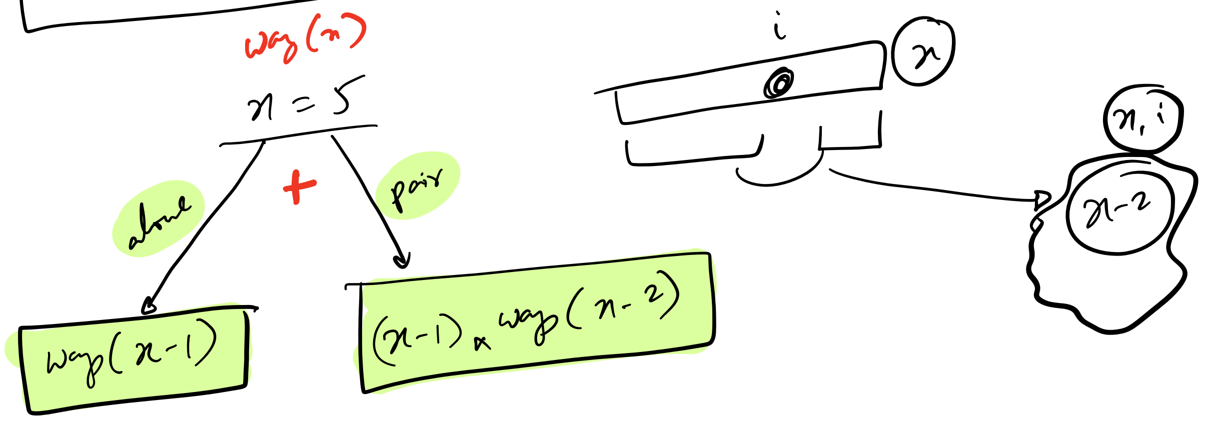
$N \geq 1$

N=1 ((1)) → 1

N= 2 ((1) (2)), ((1,2)) → 2

N = 3 ((1) (2) (3)), ((1,2),(3)), → 4
((1,3) (2)), ((2,3) (1))

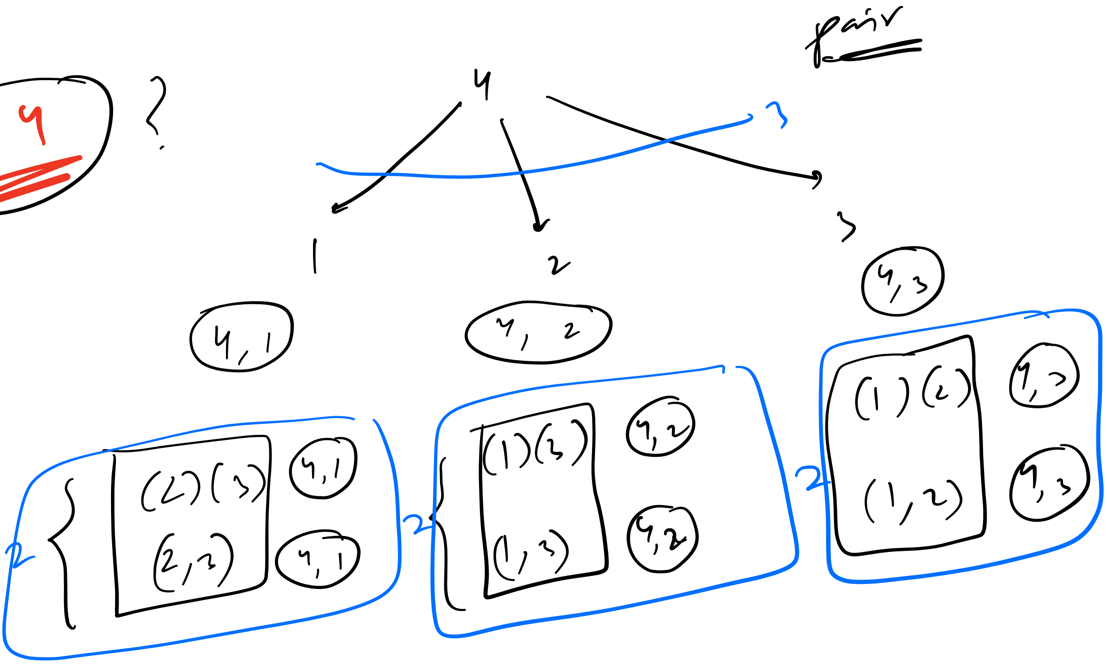
$$\boxed{\text{ways}(n) \longrightarrow \#\text{ ways of } n \text{ people to party}}$$



$N = 2$ $\quad ((1)(2))$, $((1,2))$

$N = 3$ $\quad ((1)(2)(3))$, $((1,2),(3))$,
$((1,3)(2))$, $((2,3)(1))$

$\underline{N = 4}$?



$$way(4) = \underline{(4-1)} \times \underline{way(2)} = 6$$
$$3 \times 2$$

RR

$$\boxed{way(n) = way(n-1) + (n-1)\, way(n-2)}$$

[0-N]

```
int dp[N+1] = {-1};

int wap( int n) {
    if (n <= 2) ret n;

    if(dp[n] != -1){
        ret dp[n];
    }
    ans = wap(n-1) + (n-1)* wap(n-2);
    dp[n]= ans;
    ret ans;
}
```

BC

N=1 → 1

N=2 → 2

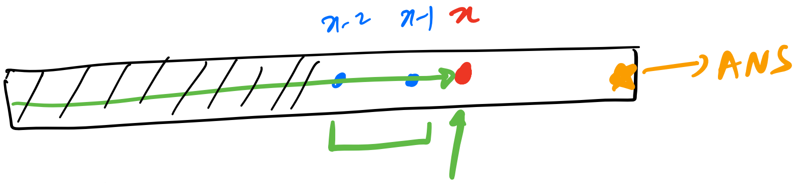
#US → N

TRPS → O(1)

$\boxed{TC = O(N)}$

$\boxed{SC = O(N)}$

Bottom Up A:-



```
int dp[N+1];
 dp[1]=1,   dp[2]= 2;
 f( i=3 ;  i<=N; i++) {
       dp[i]=  dp[i-1]+ (i-1) dp[i-2];
 }
 ret dp[N];
```

$\boxed{TC = O(N)}$ 2

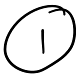
$\boxed{SC = O(N)}$

Use 2 variables like fibonaccis

$\boxed{SC \rightarrow O(1)}$

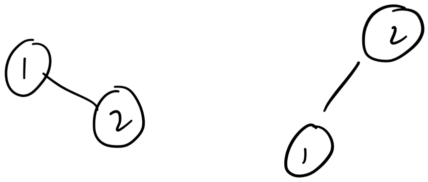
Q Given N DISTINCT KEYS.
Find the # of BSTs you can form using these!

N = 1



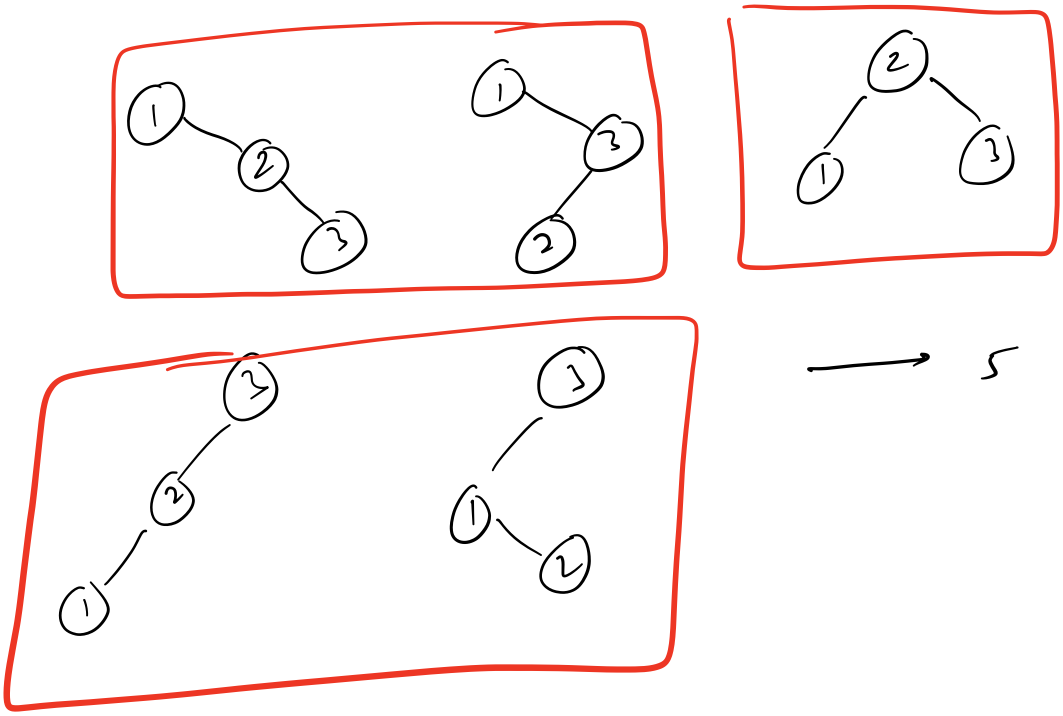
→ 1

N = 2



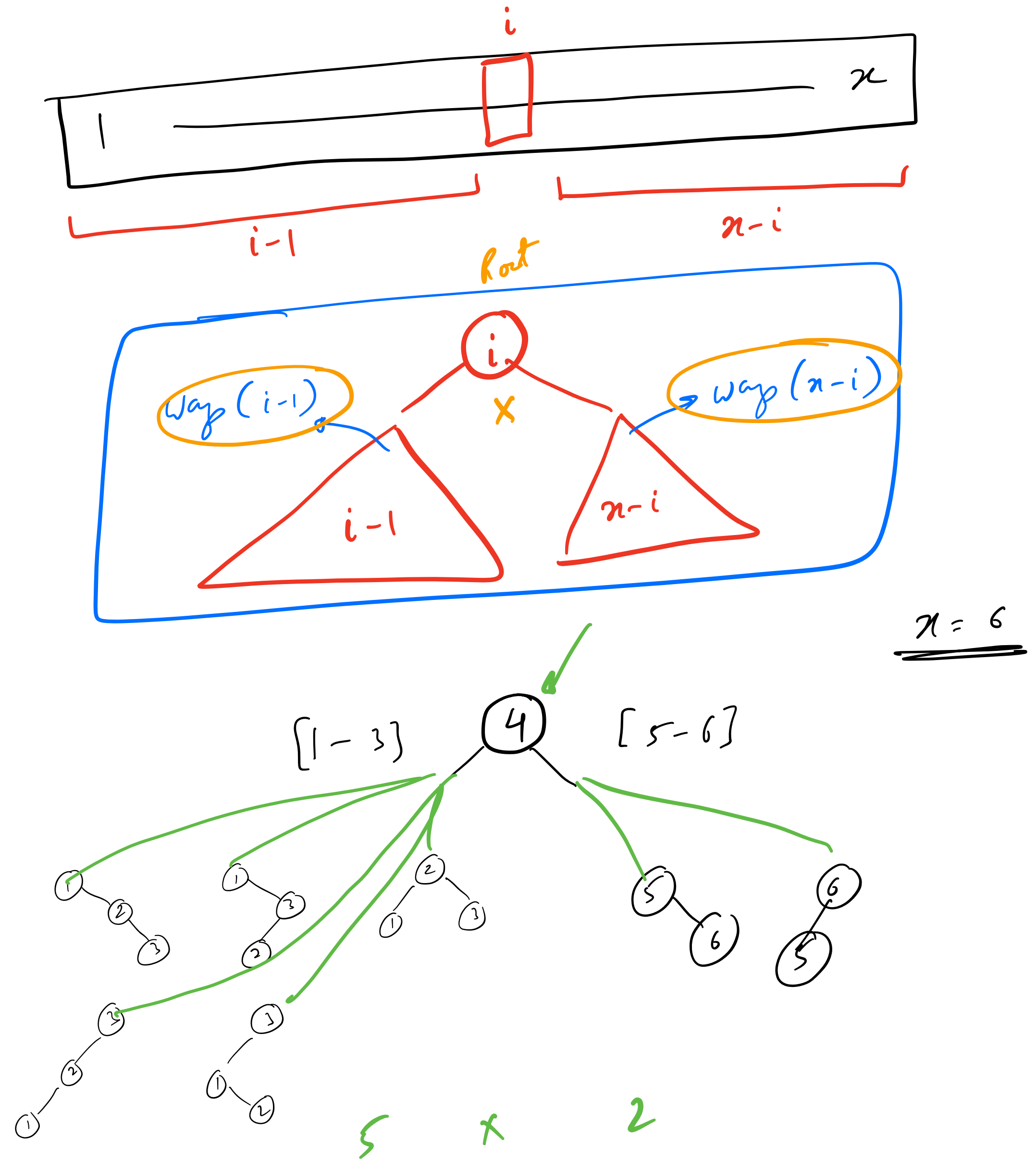
→ 2

N = 3



→ 5

ways(n) → # of ways of forming BSTs with n keys



5×2

$= 10$

$\text{way}(n) : i \text{ is root} = \text{way}(i-1) \times \text{way}(n-i)$

RR

$$\boxed{way(x) = \sum_{i=1}^{i=x} way(i-1) \times way(x-i)}$$

$x \geq 1$

$\boxed{0-N}$

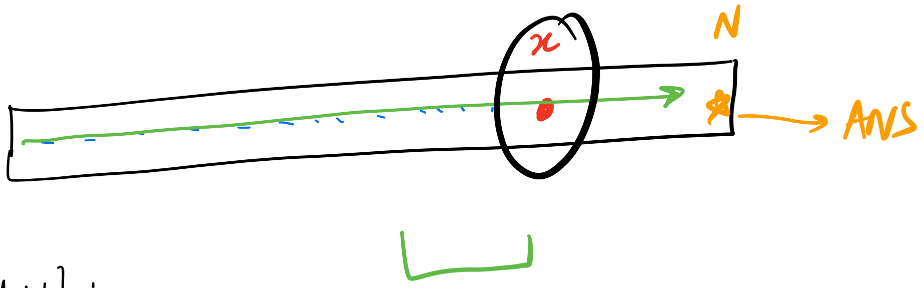
```
int dp[N+1] = {-1};
int way( int n) {
    if(n <= 1) ret 1;


    if( dp[n] != -1)
        ret dp[n];


    ANS = 0;
    f( i=1; i<=n; i++) {
        ANS += way( i-1) x way( n-i);

    }
    dp[n] = ANS;
    ret ANS;
}
```

#US → N

TRPS → O(N)

$\boxed{TC = O(N^2)}$ $\boxed{SC = O(N)}$

Bottom Up



```
int dp [N+1];
    dp[0] = 1;
  f(x=1 ; x<=N; x++){
        ANS = 0;
        f( i=1; i<=x; i++){
            ANS += dp[i-1] x dp[x-i];

        }
        dp[x] = ANS;

  }
  ret dp[N];
```

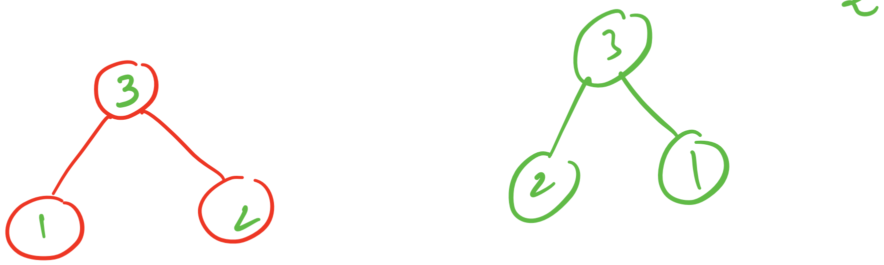
$$\boxed{TC = O(N^2)}$$

$$\boxed{SC = O(N)}$$

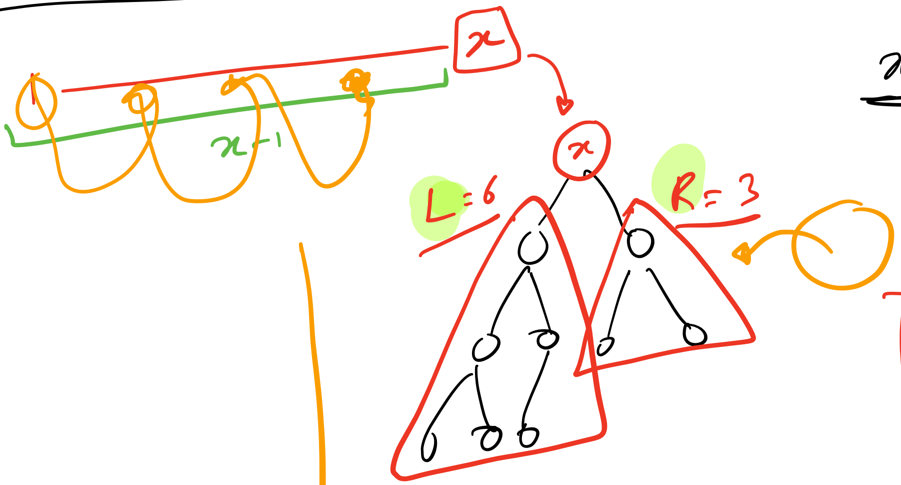
Q. Given N distinct No's.
find the # of ways of forming MAX heap!

N = 3

1, 2, 3



$$\boxed{\text{ways}(n) = \#\text{ of ways of forming max heaps with } n \text{ values}}$$



x = 10

L + R = x - 1

R = x - 1 - L

$$\boxed{{}^{n}C_{r} = {}^{n}C_{n-r}}$$

$$\boxed{\text{ways}(n) = {}^{x-1}C_{L} \times \text{ways}(L) \times \text{ways}(R)}$$

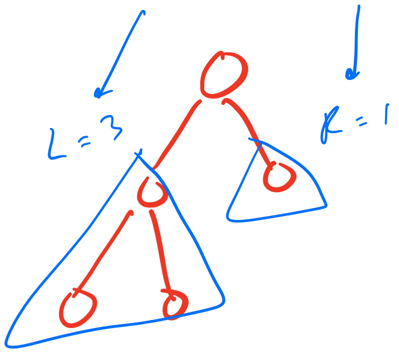
$${}^{x-1}C_{L} \rightarrow {}^{x-1}C_{x-1-L} \rightarrow {}^{x-1}C_{R}$$

RR

$$\boxed{\text{way}(n) = {}^{x-1}C_{R} \times \text{way}(L) \times \text{way}(R)}$$ ✓

0-N

X = 5

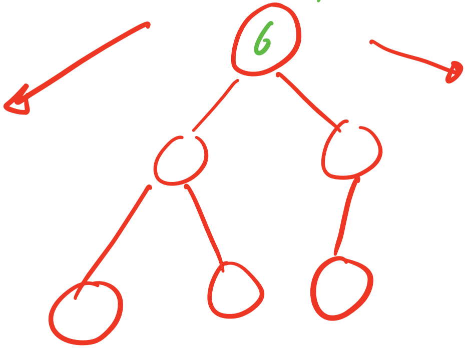


X = 6

1 2 3 4 5 6

L = 3
R = 2

1, 2, 3
1, 2, 4
1, 2, 5
1, 3, 4
⋮



4, 5
3, 5
3, 4
2, 5

$${}^{6-1}C_{3} \times \text{way}(3) \times \text{way}(2)$$

HW
DP !

Q

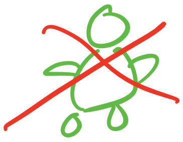
Fractional Knapsak

You have a bag with capacity C

—— N items

↳ 1. price p[i]

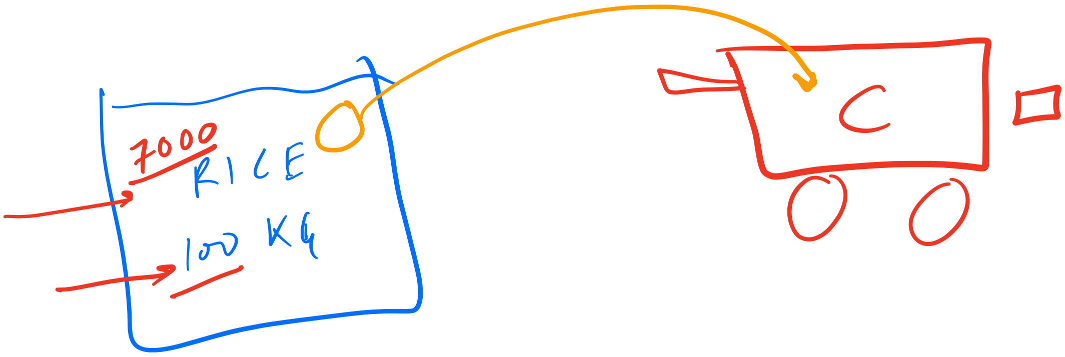
↳ 2. weight w[i]



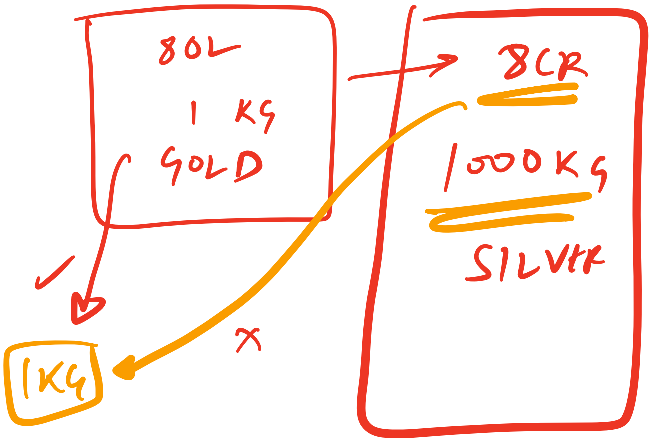
Find out the largest price you can have of the the items collected in the bag!

→ You can pick fraction of items!

DAL
RICE
WHEAT
0 1 2
i



Greedy!



Q

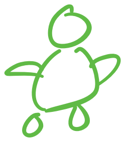
0/1 KNAPSACK

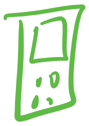
take it or leave

You have a bag with capacity ___ N items

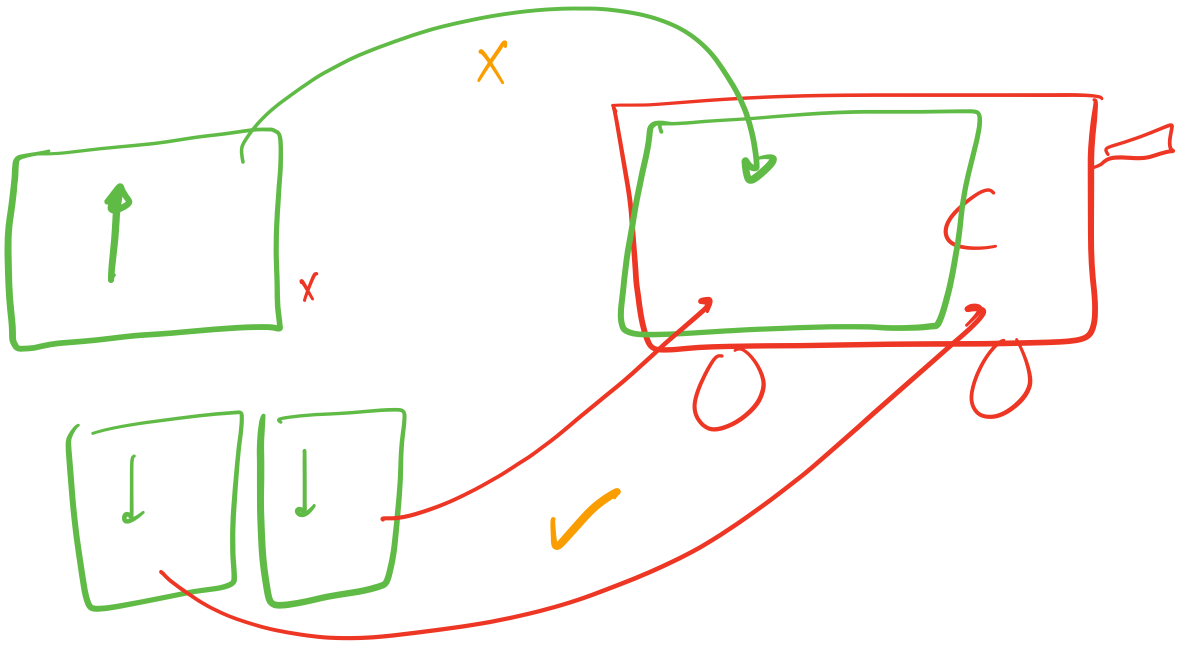
→ 1. price P[i]

→ 2. weight w[i]



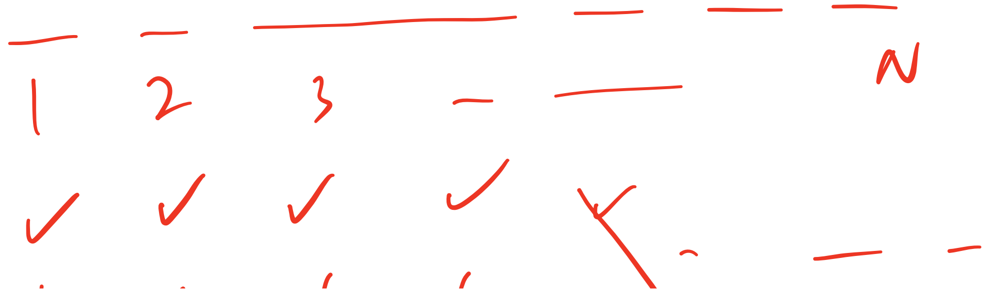


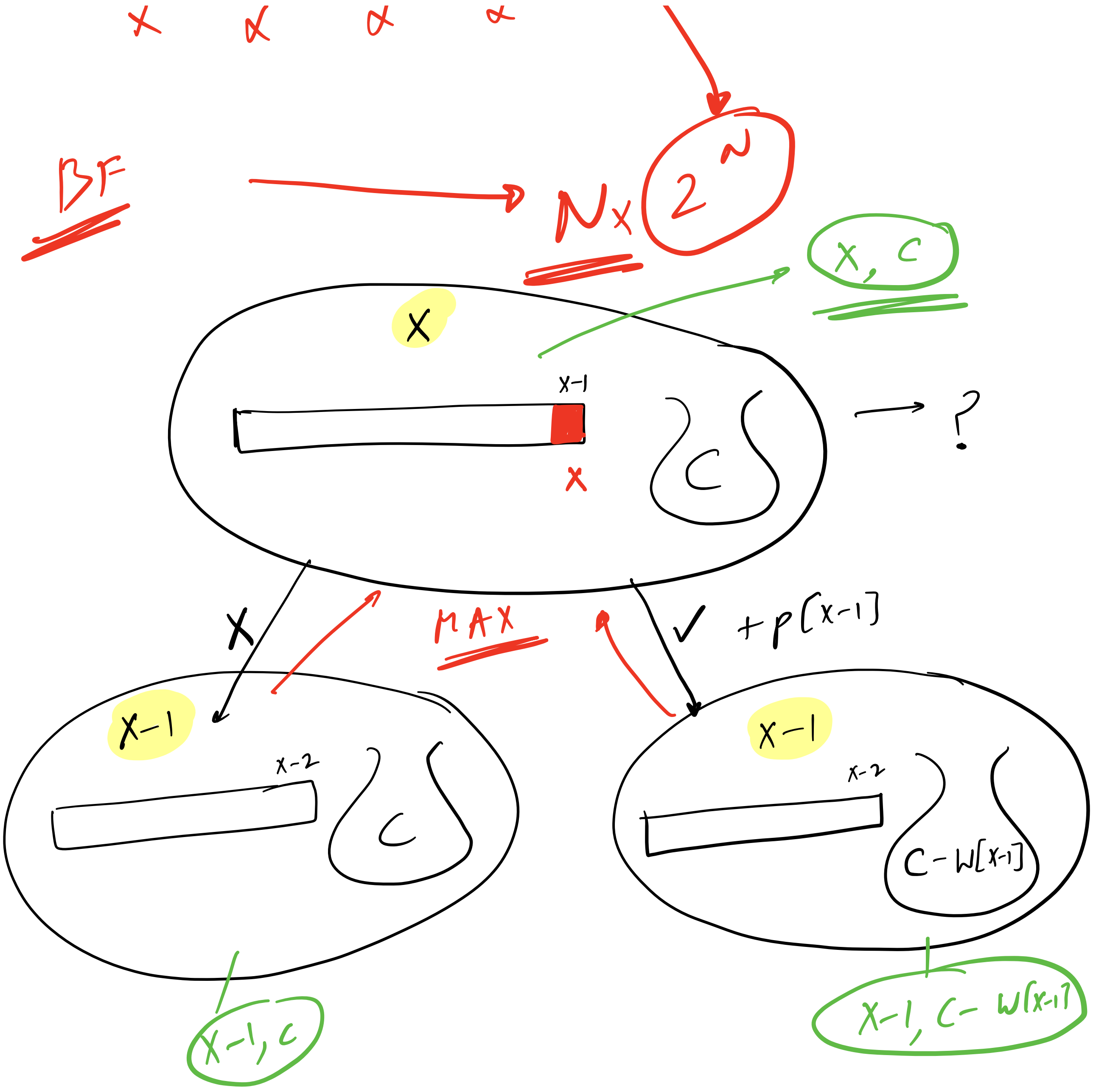
Find out the largest price you can have of the the items collected in the bag!



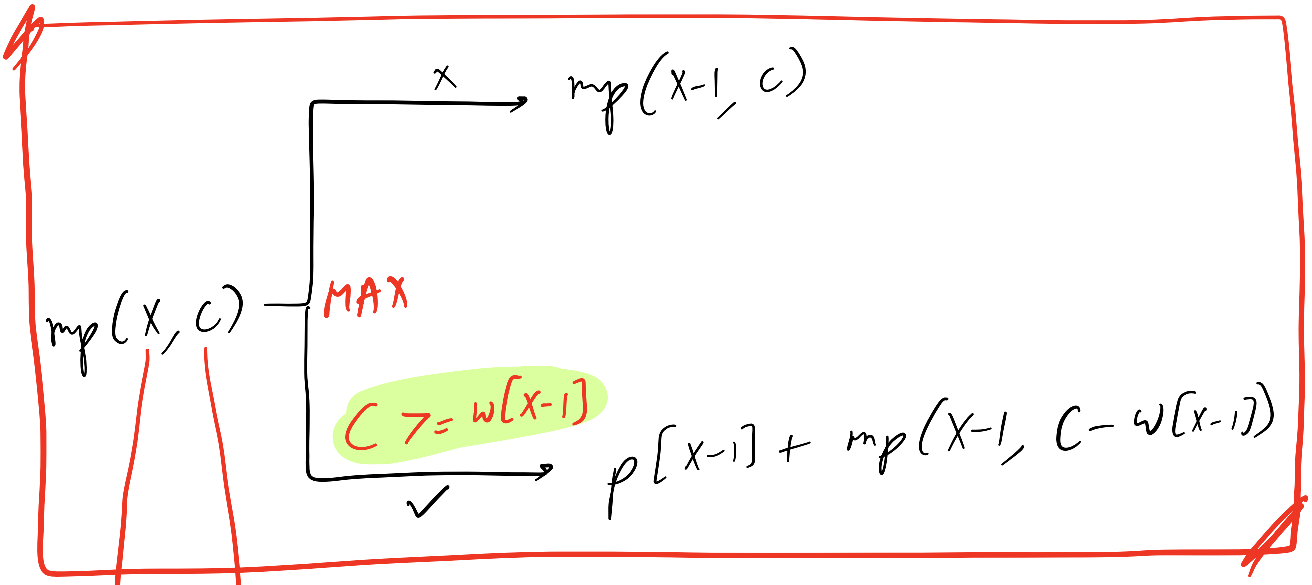
I) Generate all subsets

→ find the VALID ones & take MAX of total price!





mp(x, c) → the max price you can get considering the items [0 — x-1] & rem. cap of the bag as c



(1–N) x (0–C) → N x C UNIQUE STATES

```
int dp[N+1][C+1] = {-1};

 int mp( x, c) {
     if(x == 0) ret 0;


     if(dp[x][c] != -1){
            ret dp[x][c];
     }
     ANS = mp(x-1, c);
     if( c >= w[x-1]) {
            ANS = max( ANS,  mp(x-1, C - w[x-1]) + p[x-1]);
     }
```

```
    dp[x][c] = ANS;
    ret ANS;
}
```

#US → NC

TRPS → O(1)

$\boxed{TC = O(NC)}$

$\boxed{SC = O(NC)}$

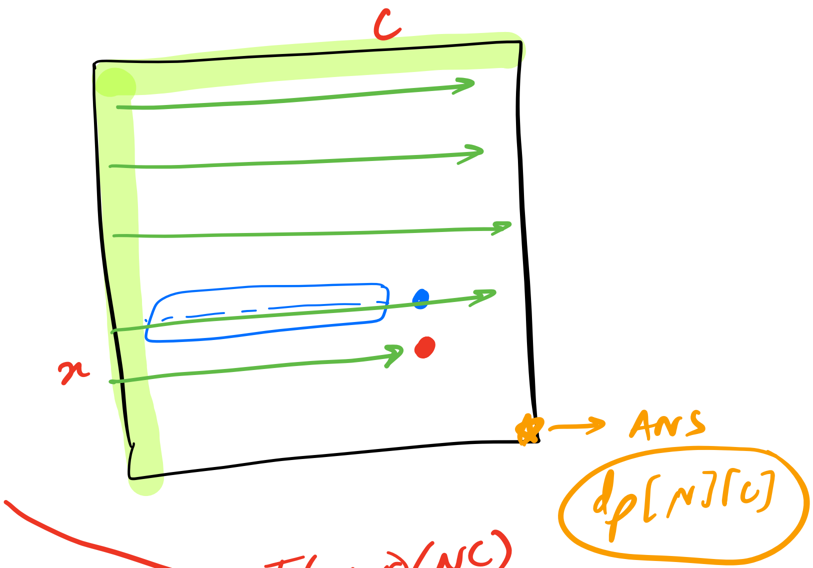
(4) Bottom Up

f ___x___

f ___c___

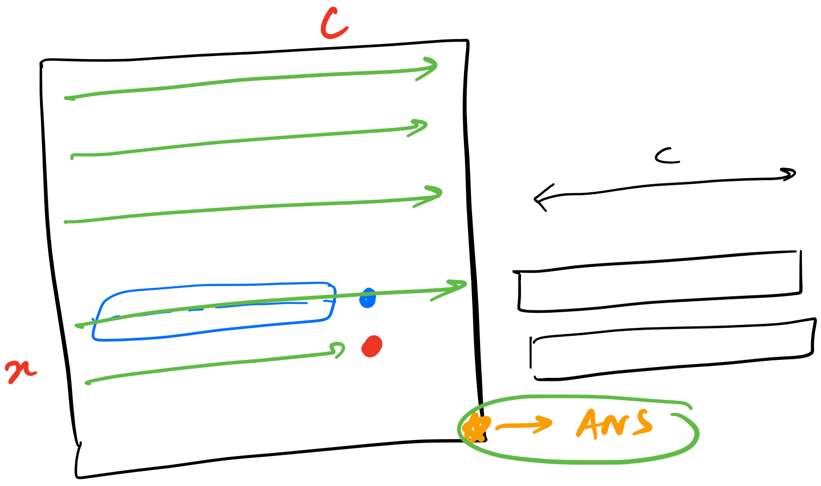
RR

dp . .



TC = O(NC)

SC = O(NC)



$\boxed{SC = O(c)}$