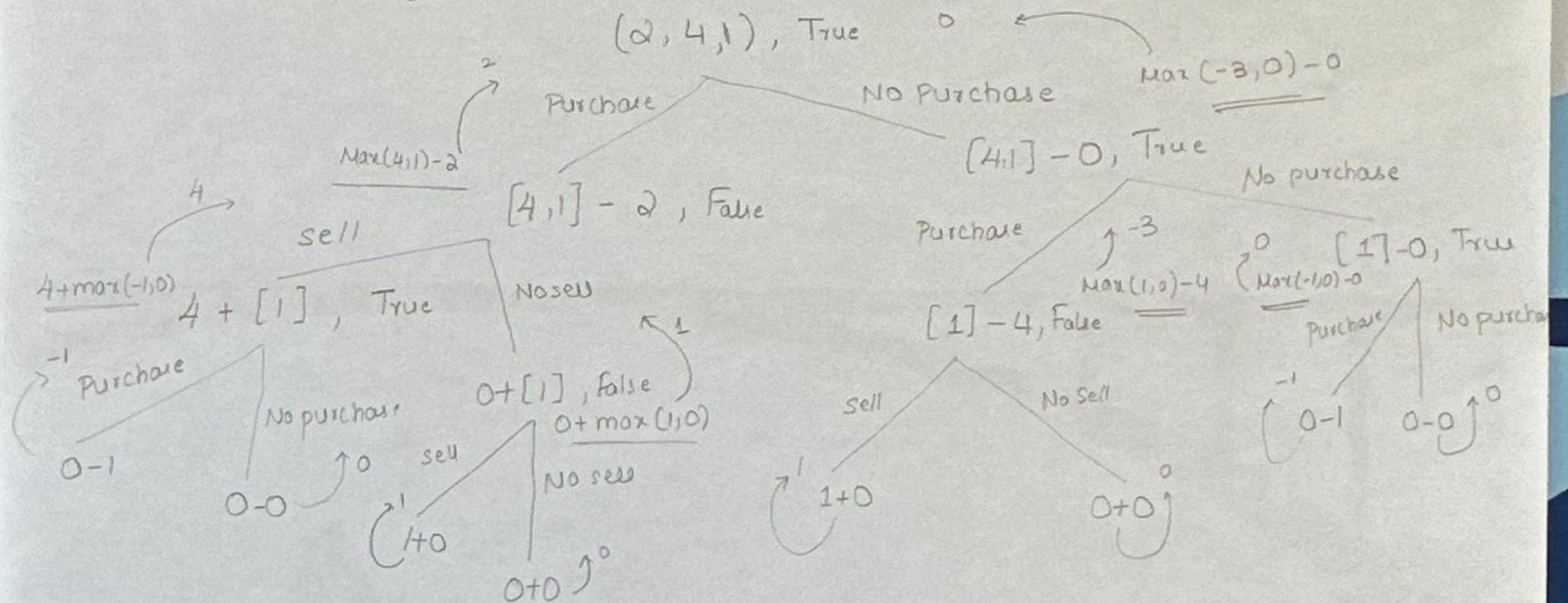


Buy & sell - 2

True : I donot hold any stock or I have not purchased any stock previously, so I can buy a stock

False : I need to sell because, I have a stock that is previously purchased & In order to buy new stock, I'll have to sell old stock $\rightarrow \text{Max}(0,0) = 0$



$[2, 4, 1]$

Buy

0 = True 1 = False

	2	4	1	0
2	$\frac{4-2}{3-0} \mid 2$	$\frac{2-3}{0+4} \mid 4$		
4	$\frac{1-4}{0-0} \mid 3$	$\frac{4+0}{0+1} \mid 4$		
1	$\frac{0-1}{0-0} \mid 0$	$\frac{1+0}{0+1} \mid 1$		
0	0	0	0	

prices

Buy \rightarrow True | 0

Purchase =

$$dp[i+1][\text{False}] - \text{prices}[i]$$

No purchase

$$dp[i+1][\text{True}] - 0$$

return $dp[0][0]$

Can't Buy : Sell \rightarrow False | 1

Sell =

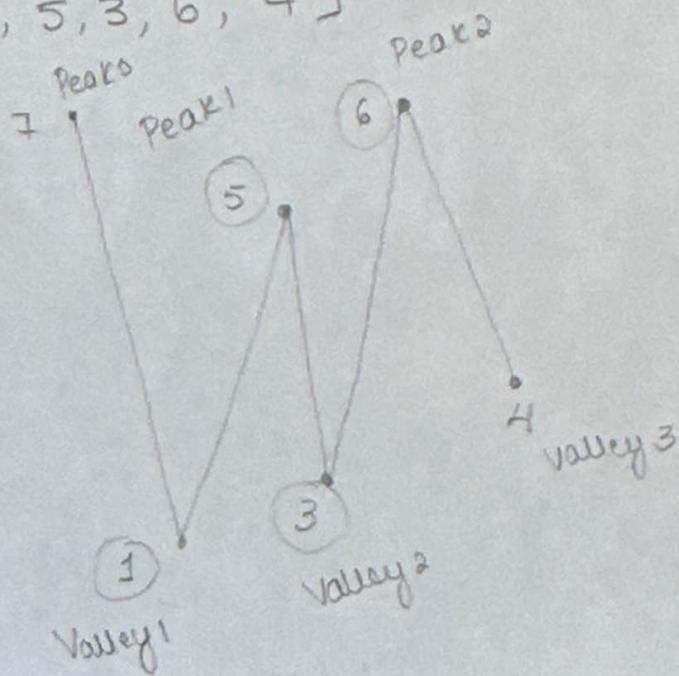
$$\text{prices}[i] + dp[i+1][\text{True}]$$

can't sell

$$= 0 + dp[i+1][\text{False}]$$

Greedy : For Every Valley check for its peak

$[7, 1, 5, 3, 6, 4]$



Combo

valley 1 \rightarrow peak 1

valley 2 \rightarrow peak 2

* peak 0 has no valley before it so profit = 0

* valley 3 has no peak after it so profit = 0

while $i < n$: # consider i to be valley

$j = i + 1$

Find the peak

while $j < n$ & $\text{price}[j] > \text{price}[j - 1]$:

$j += 1$

Profit += $\text{price}[j - 1] - \text{price}[i]$ # Peak - Valley
buy - sell

$i = j$

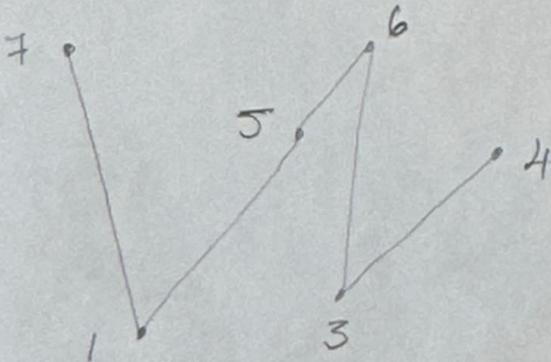
$$= 5 - 1 = 4$$

+

$$6 - 3 = 3$$

7

Optimized Greedy :



Normal Greedy

$$6 - 1 = 5$$

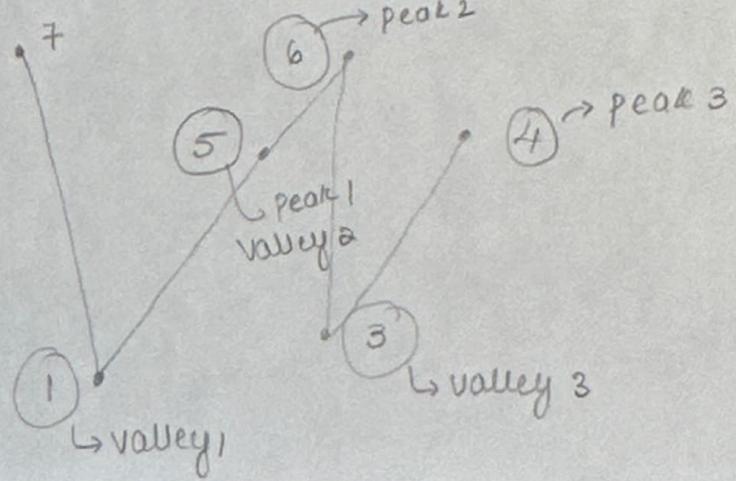
+

$$4 - 3 = 1$$

6

In optimized Greedy, we can break
 $6 - 1 = 5$, to

$$\begin{aligned} 5 - 1 &= 4 \\ + 6 - 5 &= 1 \\ \hline 5 \end{aligned}$$



```
for i in range(1,n):
    # keep moving the peak valley
```

```
if price[i] > price[i-1]:
```

```
    profit = price[i] - price[i-1]
    # sell - buy
```

```
Total profit += profit
```