M	T	W	T	F	S	S
Page No.: 95			YOUVA			
Date:					YOUVA	

## # Lecture 28: Stocks I

We are given an array prices where price of stocks are given, we can at a time buy one stock and we have to sell it before buying another stock we can report this process as many times as want, to manimise the read profit

prices [] = [7, 1, 5, 3, 6, 4]

Output = 7 ((1,5)+(3,6)

Now try to understand and observe that every stock we have two option lick | Not pick. But we need an entra variable which will tell us, if we can by buy the current stock on that particular day or not

We have to maximize our output, Thus we take

maximum of all cases.

SUBTRACT that stock's price from as ans. And we'll

ADD when we sell.

Mointain a tog buy' variable if buy = 1, purchase it or ignore it, else buy = 0 sell it or Don't sellit

Base if (ind == n), then irrespective we have bought of sell the stock, 're durn o' for no loss.

Note: After every call keep updating the buy' variable, accordingly.

Page No.: 96  Pate: YOUVA
# Memoisation I would : 15 miles it
int f (int ind, int boy, vector sint > 2 arr, vector svector sint > > k dp) {
if (ind == axx.size())  xe durn 0;  if (dp Gind J C buy J != -1)  se durn dp Gind J C buy J;
if (buy==1)  dp[ind][buy]= man(-axx[ind] + f(ind+1,0,arr,dp),  f(ind+1,1,arr,dp));  else  dp[ind][buy]= man(axx[ind]+f(ind+1,1,axx,dp),  f(ind+1,0,axx,dp);
return deplind. I buy ];
Jime Complexity = O(Nx2)
Space Complexity = O(NX2) + O(N)
- # Tabulation. Same procedures O Base Case Manipulation
Same procedures () Base (ase Manipulation.  (2) Wride variables (i, buy)  (3) Copy Recurrence.
Control of the state of the sta

-		M T W T F S S Page No.: 97 Page No.: 97 VOUVA
1 1 1 1 1 1 1 1 1 1	int solve ( vector cint > arr) int n = arr. Size(); vector cuector cint >> dp(r) dp(n)I(o) = o, dp(n)	1+1, vecdox xint > (2,0);
1111	fox(int i=n-1; i>=0; $fox(int j=0; j<$ $if(i==1)$	i++) { =1; j++) {
E	else profit = man	(axx(i)+ dp(i)(i), dp(i)(o);
Ë	dp(i]Cj]= prof	it is harden as harden as it
9	3 redura de CoJCiJ;	at his time and
	Space Comp -> O(NX2)	and an any colors
	We can space opdimize it	by using prev and curr.

	Page No.: Of YOUVA
# Lecture 29: Stocks III	
The question is enactly the same as la the only difference is that are can so (buy stocks at man 2' times.	and Sell
That means we are allowed to perform transaction = By (Buy And Selling) of	
Thus we can use the Same and just variable like the knapsack problem. To No of Transactions made.	
Try to understand only the case: When bought a stock, and when we are planni only then it is counted as A TRANSACTION	ng to sell it
Base (ase will add one more case who That means we have reached the limit and thus we can redurn 0;	t do purchase
Space Complexity = O(Nx2x3) + O(N)  Auxiliary of	Hack Space.
# Memoization.  Greate a delICICI de store ans	•
	•

	Page No.: 90
int f (int ind, int buy, int cap, vector vector)	or kint > Larr, dorkvector (int >>> L dp) {
if (ind==n    cap==0) xodurn 0;	
if (buy == 1)  deplind I buy I cap ] = man	(-arrand) + F(ind+1, 0, cap, arr, dp),
Cledentes to Carrie and St. Hillian	+ (ind+1, 1, descap, axx, dp));
dplind][buy][ap] = man(arx[ind]+f(in	1d+1, 1, cap-1, arr, dp), f(ind+1,0, cap, arr, dp);
3. return dp [ind][buy][cop]	j
# Tabulation.	
Same Rules: (1) Base Case.  (2) Variables (1), b (3) Copy Recurrence	
Jime Complexity - O(NX2X  Space Complexity - O(NX2X	3).

5 5 F

	Page No.: 100 YOUVA
	Date:
int Solve (vector Kint > Lars) {	to hat his but his
int n = are size(); vector evector eve dor eint >>> dp (n+1, vector	orcyochorsintzz(3.
we construction to the construction of the con	· 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 1
	•
for (int i=n-1; i>== 0; i) {	•
fux(int j= 0; j<=1; j++) {	1 20 Note 1 1 12
for £int 3=1; 3<=2; 3-	++){
it ( )	W/
doliJ(i)/27 = manl	Carreis + deplitistalis,
	· dp[i+][i][3];
Else. 130 kins	J. W. Landson
dolisa Jaz man Cax	x(i) + dp[i+](i)[g-1]
	delitiJloJ(3);
3	
3	of maker
.5	
redura deloscistas; loriginal	State where we
3 made F	Calls.
	•
If the question is moulded, and ( 'K' transactions instead of 2')	de are given
K' bonsactions instead of 2;	jast replace values
and everything else will remain the s	ane.
	0
Space optimize by using curre 303,	prev ()()
	Military of magic
	4

Page No.: \O\ YOUVA
Date:
Lecture 30 -> But & Sall & bakes with a Cooldown
Lecture 30 -> Buy & Sell Stocks with a Gooldown.
amplede Lame as Stock II git's just that often
a complère transaction, soure have to serve a
cooldown previod of 1 stock.
prices () = (1,2,3,0,3)
output = 3
prices(J = [1, 2, 3, 0, 3]) $output = 3$ $S(2-1) (cooldown [3] (0, 3)$
So, in the same code in Stock II, in the else
Condition where we complete theo TRANSACTION by
Selling we just do ind+2.
and a similar transfer to the state of the s
int f(int ind, int begy vector cint > & ara) {.
if (Ind 7 = and > change redurn 0
if (dpCind JC beny ] != -1)
xeturn dp (ind ] (buy);
the transmit and the to the company has been a
if (buy) - described
dplind Ilbay J= man(-axlind It flind+1,0, axx),
f(ind+1,1,axx);

if (bray) dplind of boyy ]= man (- asslind). applied I Chuy J= man (ava lind) + flind +2, 1, ava return dp (ind ] (buy);

# Lecture 30 -> Buy & Sell Stocks with