

12/01/22

DSML Intermediate - DSA

HW

Arrays - 2

(Q 1) Range sum

(Q 2) Equilibrium Index

(Q 3) Range sum Odd/Even

(Q 4) Special Index (Hw → we will discuss
in next class)

$$B \rightarrow \frac{\text{len}(A)}{\approx}$$

K times

$K > N$

$$K = \underline{\underline{K \% N}}$$

After every N_r
operations will repeat.

Q1
Amazon
 N, Q, arr
Inputs

Given an array of size N

and Q queries.

format

(s, e)
↓
start index
end index.

- DSA
- Problem Solving.
✓
- Coding Interview

Print sum of elements from index s to index e (included)

A: $\begin{bmatrix} -3, & 6, 2, 4, 5, 2, 8, -9, 3, 1 \end{bmatrix}$

$Q = 4$

	s	e	
1	2	3	$= 12$
2	2	7	$= 12$
3	4	8	$= 9$
4	0	0	$= -3$

$2+4+5+2+8-9 = 12$
 $5+2+8-9+3 = 9$

arr is the input array

for i in range(Q):

Python
Primer

s, e = Take input from the user.

sum=0

$s=0$
 $e=n-1$

for j in range(s, e+1): # $[s, e]$

$e-s+1$
 $= \underline{\max} n$

sum = sum + arr[j]

Q Iterations
 $i=0, 1, \dots, Q-1$
 $i=0 \quad N$
 $i=1 \quad N$
 $i=2 \quad N$
 \vdots
 $i=Q-1 \quad N$

print(sum)

Worst Case: $Tc = O(N \cdot Q) = O(Q \cdot N)$.

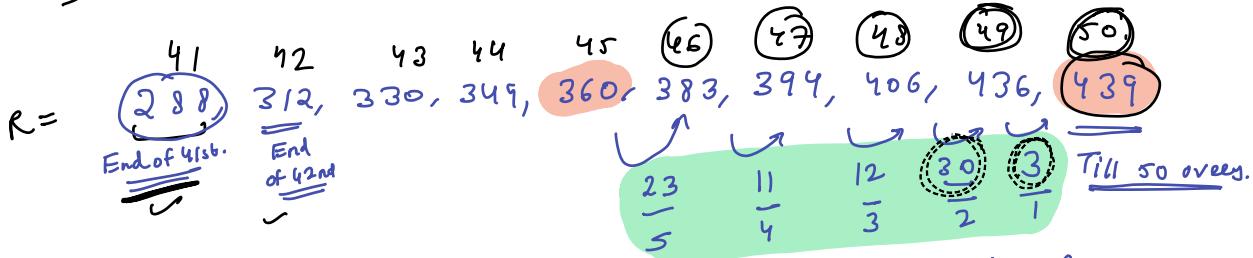
$Q \approx N$
↳

$O(N^2)$

Quadratic

ODI

Given the scores of last 10 overs of a cricket match.



$$\text{Runs scored in the last 5 overs} = R[50] - R[45] - R[46]$$

$$\underline{\underline{[46, 50]}} = 439 - \underline{\underline{360}} = 36. \boxed{79}$$

$$\text{Runs scored in the 50th over} = R[50] - R[49]$$

$$[50, 50] = 439 - 436 = 3$$

$$\begin{aligned} \text{49th over} &= R[49] - R[48] \\ [49, 49] &= 436 - 406 \\ &= 30. \end{aligned}$$

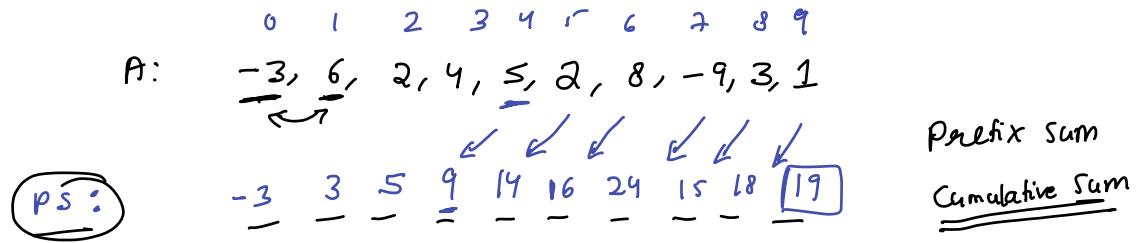
$$\begin{aligned} \text{Runs scored from } \underline{\underline{42^{\text{nd}}} \text{ to } 45^{\text{th}}} &= R[45] - R[41] \\ &= 360 - 288 \\ &= 72. \end{aligned}$$

$$\Rightarrow \underline{\underline{[s, e]}} = \underline{\underline{R[e] - R[s-1]}}$$

prefix Sum Array

Cumulative Sum $\underset{\text{Every index}}{=}$ sum of all elements from starting till the given index.

$PS[i] =$ sum of elements from index 0 to index i.



Logic:

$$ps[0] = A[0]$$

$$ps[1] = A[0] + A[1] = -3 + 6 = 3$$

$$ps[2] = \boxed{A[0] + A[1]} + A[2] = -3 + 6 + 2 = \underline{5}$$

= ps[1] + A[2]

$$ps[3] = \boxed{A[0] + A[1] + A[2]} + A[3]$$

= ps[2] + A[3] = 5 + 4 = 9

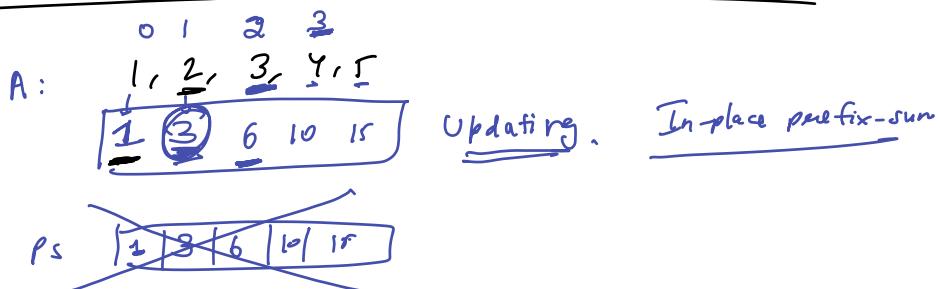
$$ps[i] = \boxed{A[0] + A[1] + \dots + A[i-1]} + A[i]$$

$$ps[i] = \underline{\underline{ps[i-1]}} + A[i]$$

Code:

```

1   → ps = [0]*N # if.
2   ps[0] = A[0] # remove.
3   for i in range(1, N):
4       ps[i] = ps[i-1] + A[i] → TC: O(N)
        # A[i] = A[i-1] + A[i] → SC: O(1)
    
```



$O(1)$ space:

for i in range($1, n$):

$$A[i] = A[i-1] + A[i]$$

Q → queries.

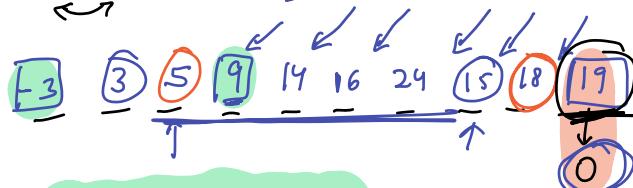
$[s, e]$

$$s = 2, e = 9$$

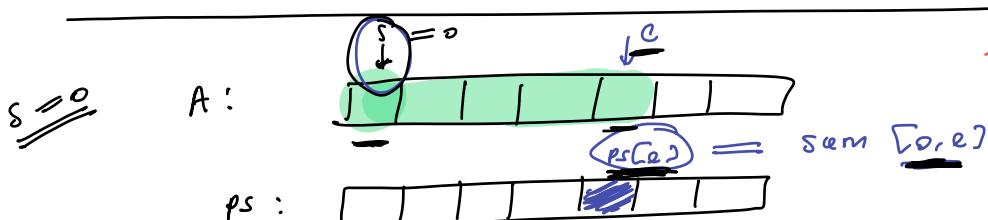
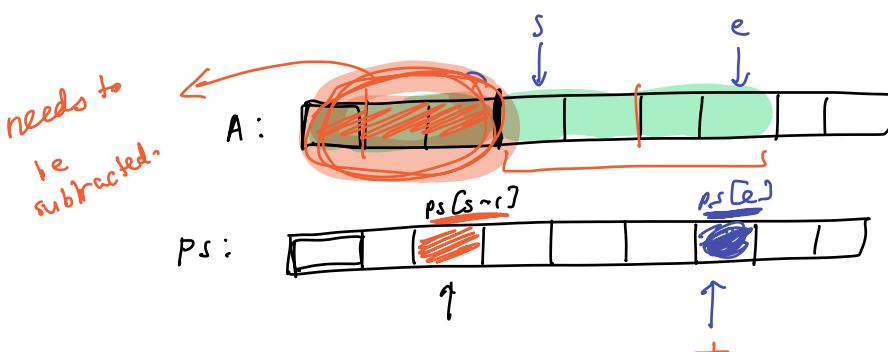
$$ps[9] - ps[1]$$

A: $-3, \underline{6}, 2, 4, \underline{5}, \underline{2}, 8, -9, 3, 1$

ps :



s	e	$ps[e] - ps[s-1]$
1	3	$= ps[3] - ps[0] = 9 - (-3) = 12.$
2	7	$= ps[7] - ps[1] = 15 - 3 = 12$ <u>$ps[-1]$</u>
4	8	$= ps[8] - ps[3] = 18 - 9 = 9$
0	0	$= ps[0] - ps[-1] = -3 - 1 = -2$
0	2	$= ps[2] - ps[-1] = 3 - 1 = 2$



Pseudo Code

arr is the input array

① # 1. Construct the ps array.

$$\Rightarrow ps = [0] * N$$

$$ps[0] = A[0]$$

for i in range (1, N):

$$\Rightarrow ps[i] = \underline{ps[i-1]} + A[i]$$

$$TC: O(N)$$

$$SC: O(N)$$

② for i in range (0, N):

s, e = Take input from the user.

$$sum = 0$$

if s == 0:

$$sum = \underline{ps[e]}$$

else:

$$sum = \underline{ps[e]} - \underline{ps[s-1]}$$

$$TC: O(1)$$

$$SC: O(1)$$

print(sum)

$$\text{Total TC: } O(N+Q) = O(\max(N, Q))$$

$$\text{Total SC: } O(N+1) = O(N)$$

$$\begin{cases} c = 2N \\ Q = N^2 \end{cases}$$

$$O(N+N^2) = O(N^2)$$

Lower order
constants
will be dropped.

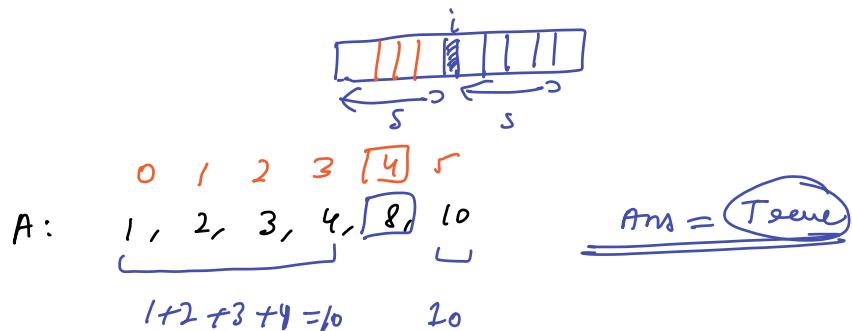
5 minutes break

Ques-2

Direct - I

Given an array, Return True if there exist an equilibrium index in the array.

$EI \Rightarrow$ index for which
sum of elements on left side of i = sum of all the elements on the right side of i



Direct

$$\begin{array}{ccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ -7, 1, 5, & \underbrace{2,}_{-4,} & 3, 0 & \end{array}$$

$$\begin{aligned} -7 + 1 + 5 &= -1 \\ & \\ -4 + 3 + 0 &= -1 \end{aligned}$$

$$\boxed{S_L(i) = ps[i-1]}$$

\downarrow

Or, $(i-1)$

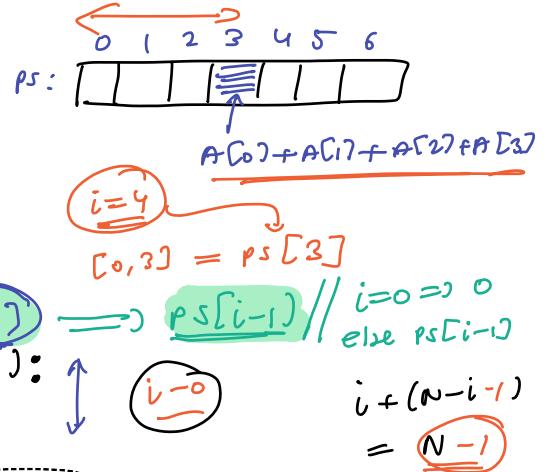
NOTE 1: For index $i=0$: $S_L = 0$

NOTE 2: For index $i=(N-1)$: $S_R = 0$

Brute Force

Check for all indices $i = 0 \rightarrow \underline{N-1}$
if this is an eq^m index.

$$\sum [0, \underline{i-1}] = \sum [\underline{i+1}, \underline{N-1}]$$



def check EI (arr) :

n = len(arr)

for i in range (n) :

$sl = 0 \# [0, \underline{i-1}]$

for j in range (0, i) :

$sl += arr[j]$

$sl = 0 \# [\underline{i+1}, \underline{N-1}]$

for j in range (i+1, n) :

$sr += arr[j]$

if $sl == sr$:

return True

return False.

Sum from $[0, e]$
 $= ps[e]$

$[0, \underline{i-1}]$
 $= ps[\underline{i-1}]$

a) sum from $\underline{(i+1)}$ to $\underline{(N-1)}$

Range sum from $[s, e] = ps[e] - ps[s-1]$
if $s == 0 \Rightarrow ps[0]$

$\underline{(i+1)}, \underline{(N-1)} ? = ps[N-1] - ps[i+1-1]$
 $= ps[N-1] - ps[i]$

```

def checkEI(arr):
    n = len(arr)
    ps = [0] * n
    ① # Build ps array
    for i in range(n):
        # sl=0 # [0, i-1]
        sl = ps[i-1] # for i=0, sl=0 => O(1)
        sr = ps[N-1] - ps[i] # for i=N-1, sr=0. => O(1).
        if sl == sr:
            return True
    return False.

```

TC: $O(N)$, SC: $O(N)$

$\xrightarrow{\quad}$

$\begin{cases} \text{range sum} \\ \text{range sum.} \end{cases}$

Total: $TC = O(N) = \text{Build} + \text{Find EI}$
 $SC = O(N) \qquad \qquad \qquad N + N = \textcircled{3}N$

HW: Try to do using $O(1)$ space

Ques \rightarrow Sum of all elements from index i to j

if $i \neq 0$: $sum = ps[j] - ps[i-1]$

if $i == 0$: $sum = ps[j]$

Ans \rightarrow

$$i = 3$$

$$sum = \underline{ps[7]} - \underline{ps[2]}$$

$$j = 7$$

Ques 3 :

Given array & Q queries.

Google
CodeNation
JP Morgan

A ← format

s e 'O' \Rightarrow sum of all odd index elements from \checkmark PS_o.
[s, e]

s e 'E' \Rightarrow sum of all even index elements from \checkmark PS_e.
[s, e].

0 1 2 3 4 5 6 7
A: 2, 3, 1, -1, 0, 8, 5, 4

Q = 2 $\xleftarrow{[1-5]}$

S	e	O/E
3	6	O
(5	E	= $A[3] + A[5] = -1 + 8 = 7$

$\Rightarrow A[2] + A[4] = 1 + 0 = 1$

Range sum \Rightarrow Prefix sum. \Rightarrow sum of all index elements from [0, i]

A: 2, 3, 1, 6, 4, 5 $\quad s = 1, \quad e = 5$

$$PS_e[5] - PS_e[0] \\ = 7 - 2 = 5$$

$PS_e[i] = \boxed{2} - \boxed{2} \quad 3 \quad 3 \quad 7 \quad \boxed{7}$
 $\boxed{PS[e] - PS[s-1]}$
sum of all even index elements from [0, i]

Ques 2 -

0 1 2 3 4 5 6
4, 1, 0, -2, 3, 2, 5

$$s = 2$$

$$e = 6$$

$$PS_e = [4, \boxed{4}, 4, 4, 7, \boxed{7}, \boxed{12}]$$

$$PS[e] - PS[s-1]$$

$$= 12 - 4 = 8.$$

$$PSe = [0] \times n$$

$$PSe[0] = A[0]$$

for i in range $(1, n)$:

if $i \% 2 == 0$:

even

$$PSe[i] = PSe[i-1] + A[i]$$

else:

$$PSe[i] = \underline{PSe[i-1]}$$

$$PSO = [0] \times n$$

$$\underline{PSO[0]} = 0 \quad \# 0 \text{ is not odd}$$

for i in range $(1, n)$:

if $i \% 2 == 0$:

$$\underline{PSO[i]} = \underline{PSO[i-1]}$$

else: $\# \text{odd}$

$$\underline{PSO[i]} = \underline{PSO[i-1]} + A[i]$$



Ans.

$$PSO[i]$$

$$0, \underline{4}, \underline{4}, \underline{5}, 5$$

$$s=1$$

$$c=3$$

$$\Rightarrow PSO[3] - PSO[0]$$

$$\Rightarrow 5 - 0$$

$$= s$$

\Leftrightarrow Special Index

$$\bullet PSO$$

$$\bullet PSe$$

Doubts

$$A: 2, 3, 1, 6, 4, 5$$

✓ $A_e : 2^{\frac{1}{2}}, 0^{\frac{1}{3}}, 1^{\frac{1}{1}}, 0^{\frac{1}{6}}, 4^{\frac{1}{4}}, 0^{\frac{1}{5}} \Rightarrow 2, 2, 3, 3, 7, 7 \quad (PSe)$

✓ $A_o : 0^{\frac{1}{2}}, 3^{\frac{1}{3}}, 0^{\frac{1}{1}}, 6^{\frac{1}{6}}, 0^{\frac{1}{4}}, 5^{\frac{1}{5}} \Rightarrow 0, 3, 3, 9, 9, 14 \quad (PSe)$

Chaitanya:

Book: Element of programming Interviews in Python.

Jagrut:

Practice, patterns - Observation

Sushant:

TLE.

Ravi:

Write good and optimized.

Venkatesh Chiranjeevi:

Implementation: . Try to read others' code.

. Read documentation

. Pseudo Code before writing actual code.

Anshu: ⑥ Class

① Assignments

② HW

Mock Interviews

: Interviews

Narendra:

Optional Problem
Solving session

Why TLE?