Subarrays basics

Printing subarrays

Generating all subarrays sum

Printing all subarray sums

Announcement : 9 June 9:00 – 10:30

1.5 hours

3 questions

you should be able to solve

atleast 2

→ Watch all lectures if not already

→ complete HWs & assigments

Sylabus : Arrays & TC

Constest discussion on same day

Subarray basics

- Continous part of an array is called subarray
- Single element is subarray ? Yes
- Complete array is subarray ? Yes

  0 elements " " ? No.

	0	1	2	3	4	5	6	7	8	9
Ar[10] :	-2	4	6	3	8	1	4	3	2	-10

{2 4 1 6 -3 7 8 4}

{1 4} ✗   {4 1} ✓

{1 6 8} ✗

{6 1 4 2} ✗

{7 8 4} ✓

{4 5 1 9 0 2 3 5}

{5} ✓

{4 5 1 0} ✗   a c d

{9 0 2 3} ✓

{4 5 1} ✓

Subarray can be defined by a start ind and end index

s   e $\boxed{s \leq e}$

$$\begin{array}{cccc} 0 & 1 & 2 & 3 \\ \{4 & 5 & 1 & 3\} \end{array}$$

$s = 0 \quad e = 2 \quad \{4\ 5\ 1\}$

$s = 0 \quad e = 0 \quad \{4\}$

$s = 3 \quad e = 1 \quad \not\prec$ invalid

$$\boxed{A\ [s : e] \qquad \text{len} : \quad e - s + 1}$$

Number of subarrays

$$\begin{array}{ccccc} & 0 & 1 & 2 & 3 \\ A[4] = \{ & -1 & 3 & 2 & 3 \; \} \end{array}$$

$\uparrow s$ $\qquad s \le e$

s	e		s	e		s	e		s	e	
0	0	{-1}	1	1	{3}	2	2	{2}	3	3	{3}
0	1	{-1 3}	1	2	{3 2}	2	3	{2 3}			
0	2	{-1 3 2}	1	3	{3 2 3}						
0	3	{-1 3 2 3}									

Ans: 10

$A[N] = \{ \overset{0}{a_0} \quad \overset{1}{a_1} \quad \overset{2}{a_2} \quad \ldots \quad \overset{N-1}{a_{N-1}} \}$ $\qquad s \le e$

No. of subarrays:

s	e	s	e		s	e
0	0	1	1		N-1	N-1
0	1	1	2			
⋮		⋮				
0	N-1	1	N-1			
N		N-1			1	

$$N + (N-1) + (N-2) \ldots 1 = \frac{N \times (N+1)}{2} \quad ✓$$

Q1) Given a subarray, print it. A, s, e

$s = 1 \quad e = 3$

0	1	2	3	4
2	3	4	5	7

$\{ 2 \quad \boxed{3 \quad 4 \quad 5} \quad 7 \}$

```
printsub (A, s, e) {
    for (i=s; i<=e; i++) {
        print (A[i])
    }
}
```

$\boxed{\text{TC: } O(N)}$

Q2) Given N array elements, print start and end index of each and every subarray?

$s \leq e$

	0	1	2	3	
A: {	6	8	-1	7	}

TC: $O(N^2)$

s	e
0	0
0	1
0	2
0	3
1	1
1	2
1	3
2	2
2	3
3	3

```
for(s=0; s<=N-1; s++){
    for(e=s; e<=N-1; e++){
        print(s, e)
    }
}
```

Q2) Given N array elements, print each and every subarray?

	0	1	2	3	
A: {	6	8	-1	7	}

CANNOT OPTIMISE

$\boxed{TC: O(N^3)}$

s	e	
0	0	{ 6 }
0	1	{ 6 8 }
0	2	{ 6 8 -1 }
0	3	{ 6 8 -1 7 }
1	1	{ 8 }
1	2	{ 8 -1 }
1	3	{ 8 -1 7 }
2	2	{ -1 }
2	3	{ -1 7 }
3	3	{ 7 }

```
for(s=0; s <= N-1; s++){
    for(e=s; e<=N-1; e++){
        for(k=s; k<=e; k++){
            print(A[k])
        }
        print(\n)
    }
}
```

TC to print a subarray : $O(N)$

Total No. of subarrays : $\frac{N \times (N+1)}{2} \simeq O(N^2)$

TC to print all subarray : $N \times N^2$

: $O(N^3)$

1 banana = 10Rs

N^2 bamas = $N^2 \times 10$

Q2) Given N array elements, print each subarray sum?

	0	1	2	3
A:	6	8	-1	7

s	e	subarray		sum
0	0	{6}	→	6
0	1	{6 8}	→	14
0	2	{6 8 -1}	→	13
0	3	{6 8 -1 7}	→	20
1	1	{8}	→	8
1	2	{8 -1}	→	7
1	3	{8 -1 7}	→	14
2	2	{-1}	→	-1
2	3	{-1 7}	→	6
3	3	{7}	→	7

TC: $O(N^3)$

```
for(s=0 ; s <= N-1 ; s++){
    for(e=s ; e<=N-1; e++){
        sum=0
        for(k=s ; k<=e ; k++){
            sum+= A[k]
        }
        print(sum)
    }
}
```

prefix sum

TC: $O(N^2)$

SC: $O(N)$

Build pf array → N

```
for(s=0; s<=N-1; s++){
    for(e=s; e<=N-1; e++){        → N^2
        if (s>0){ pf[e] - pf[s-1]}
        else{ pf[e] }
    }
}
```

sum(A[s:e]) =

Finding sum of subarray [s : e]

```
sum=0
for(k=s; k<=e; k++){
    sum+= A[k]
}
print(sum)
```

{ 1 2 3 4 }

pf: { 1 3 6 10 }

sum [A[s:e]] : pf[e] - pf[s-1]

Carry forward

Q Given an ar[N] print all subarray sums starting at index 3

	0	1	2	3	4	5 (e)	6	7	8	9
A[10] =	3	8	4	7	9	4	3	2	7	6
sum = 0				7	16	20	23	25	32	38

print 7 16 20 23 25 32 38

TC: $O(N^2)$

SC: $O(1)$

```
for (s=0; s<=N-1; s++) {
    sum = 0
    for (e=s; e<=N-1; e++) {
        sum = sum + A[e]
        print (sum)
    }
}
```

print all subarray sum starting from index 0

print all subarray sum starting from index 1

⋮ 2

⋮

N-1

Break (10:22 - 10:32)

Google

Q) Given N array elements, return sum of {All subarray sums}

$$A = \{\overset{0}{6} \quad \overset{1}{8} \quad \overset{2}{-1} \quad \overset{3}{7}\}$$

s	e			
0	0	{6}	→	6
0	1	{6 8}	→	14
0	2	{6 8 -1}	→	13
0	3	{6 8 -1 7}	→	20
1	1	{8}	→	8
1	2	{8 -1}	→	7
1	3	{8 -1 7}	→	14
2	2	{-1}	→	-1
2	3	{-1 7}	→	6
3	3	{7}	→	7

$\boxed{\text{sum : } 94}$

$O(N^3)$

Find all subarray sums

using global var

$O(N^2)$

prefix sum

using global var

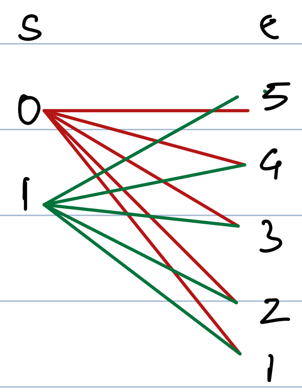
$O(N^2)$

carry forward

using global var

How many subarrays index 1 is present?

$$
\begin{array}{ccccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 \\
A = \{ & 3 & \boxed{-2} & 4 & -1 & 2 & 6 \ \}
\end{array}
$$



$5 \times 2 = 10$

$$
\begin{array}{ccccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 \\
A = \{ & 3 & -2 & \boxed{4} & -1 & 2 & 6 \ \}
\end{array}
$$

s	e
0	5
1	4
2	3
	2

$3 \times 4 = 12$

$$\begin{array}{cccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & \\ A = \{ & \boxed{3} & -2 & 4 & -1 & 2 & 6 & \} \end{array}$$

s	e
0	5
	4
	3
	2
	1
	0

$1 \times 6 = 6$

0	1	2	3	4	..	i	..	N-1	
{ a_0	a_1	a_2	a_3	a_4	....	$\boxed{a_i}$		a_{N-1}	}

s	e	Contributi
0	N-1	
1	N-2	
2	⋮	
⋮		
i	i	

$[0\ i]$ $\quad$ $[i\ \ N-1]$

$(i+1)$ $\quad$ $(N-1-i+1)$

$(i+1) \times (N-i)$

No. of subarrays having i^{th} index

	0	1	2	3
	{ 6	8	-1	7 }
$(i+1)\times(N-i)$	4	6	6	4
Contribution	6×4	6×8	-1×6	7×4
	24	48	-6	28

= 94

Contribution technique

TC: $O(N)$

SC: $O(1)$

```
sum = 0
for (i=0; i<N; i++){
    # freq of A[i]
    freq = (i+1)(N-i)
    contribution = A[i] x freq
    sum += contribution
}

return sum
```

Done!

1) Paper , pen