Congratulations:

a. Welcome to DSA & congratulations & Completing JAVA Basics & let's start of new journey on DSA to more fun & Challenging Content

Self Intro:

N. Satya Sai Siva Rama krishna

Teaching exp 3+

Work Experience in Scaler : 2+

Session will start sharp : 9:05AM

Todays Content:

a) Factors Optimization

b) Rotate arr[] from right to left

c) logarithmetic basics

d) Total DSA Content Intro

e) Habit forming Sessions

Q1) Given N, return no: of factrs of N? DSA: Intro

↳ factor? Q1: Is 4 a factr of 24: `24%4==0`

Q2: Check if i is a factr of N: `N%i==0`

Countfactrs:

Q3: N=10 factrs = {1, 2, 5, 10} ans=4

Q4: N=12 factrs = {1 2 3 4 6 12} ans=6

N=6 factrs = {1 2 3 6} ans=4

```
int countfactrs(int N){
    int c=0;
    for(int i=1; i<=N; i++){
        if(N%i==0){
            c=c+1
        }
    }
    return c;
}
```

Ass: System takes 10^8 iteratns = 1 sec ? [TC2]

i:[1...N]
Given N
N iteratns

Unitary Method:

5 apples = 60 rupes

1 apples = 60/5 = 12 rupes

8 apples = 8 * 12 = 96 rupees

10^8 iteratns = 1 sec ⇔ 1 iteratn = $1/10^8$ sec

Input N	Iterations	Execution Time
10^9	10^9 iterati	10^9 iteratn ⇒ $10^9/10^8$ ⇒ 10 sec.
10^{18}	10^{18} iteratns	10^{18} iteratn ⇒ $10^{18}/10^8$ ⇒ $10^{18-8} = 10^{10}$ = 10 sec

Note: $\frac{a^m}{a^n} = a^{m-n}$

Note: 10^{10} sec = 317 years

Observations: Duration: 9:30PM → 9:33PM

1. a <= 10 : max a = 10

2. Say i * (j) == N
   → i & j are factors of N.
   → j = N/i
   → i & N/i are factor of N

Example:

N	i	N/i	
36	4	36/4 = 9	36%9==0
12	3	12/3 = 4	12%4==0
10	2	10/2 = 5	10%5==0

If i is factor of N, N/i is also factor of N

N = 24

i		N/i
1	<=	24
2	<=	12
3	<=	8
4	<=	6
6		4
8		3
12		2
24		1

Obs: Iterate on all i values of i^nd half
i <= N/i
i * i <= N

i value
start i = 1
keep going i * i <= N

N = 36

i		N/i
1	<=	36
2	<=	18
3	<=	12
4	<=	9
6	<=	6
9		4
12		3
18		2
36		1

Obs: If we just iterate in Part 1, [1, i * i <= N] we can get all factors of N.

```
int countfactors(int N){
    int c=0;
    for(int i=1; i*i<=N; i++){
        if(N%i==0){ // i & N/i
            if(i==N/i){ c=c+1 }
            else { c=c+2 }
        }
    }
    return c;
}
```

N = 36

i	i*i<=36	i, N/i	c	i=i+1
1	1*1<=36	1, 36	+2	
2	2*2<=36	2, 18	+2	
3	3*3<=36	3, 12	+2	
4	4*4<=36	4, 9	+2	
5	5*5<=36	5 not		
6	6*6<=36	6, 6	+1	
7	7*7<=36	Break.		

```
for(int i=1; i*i<=N; i++){
```

$i^2 <= N \Leftrightarrow i <= \sqrt{N}$

$x^2 = 25$	$x^2 = 49$
$x = \sqrt{25}$	$x = \sqrt{49}$

```
i=1; i<=√N; i++) { iterations = √N iterations
```

Ass: 10^8 iterations = 1sec $\Leftrightarrow$ 1 iteration = $1/10^8$

Input N	Iterations	Execution Time
$N = 10^{18}$	$\sqrt{10^{18}}$ iterations	$\sqrt{10^{18}}$ iterations = 10^9 iterations = 10sec $= 10^9/10^8 = 10^{9-8} = 10^1 = 10$ sec

Prime Number: A number N is said to be prime if count of factors == 2

```
boolean is_Prime(int N){
    if(countfactors(N)==2){
        return true;
    }
    else { return false }
}
```

Log Basics 10:10 pm

10:20pm → 10:30pm

$\log_b^a$ = { To what power we need to raise b to get a }

$\log_2^8 = 3$, $2^3 = 8$

$\log_3^{27} = 3$, $3^3 = 27$

$\log_4^{16} = 2$ $4^2 = 16$

$\log_5^{25} = 2$ $5^2 = 25$

$\log_2^{32} = 5$ $2^5 = 32$

Few Formula

a → b^N, b → b

1. $\log_b^{b^N} = N$

2. if $N = 2^k$, $\log_2^N = k$

$\log_2^{2^k} = k$

if $N = 2^k$, $\log_2^N = k$

Doubts: 10:10pm → 10:20pm

$\log_2^{10}$ = $2^3 = 8$, $2^4 = 16$ → $2^{3.n} = 10$ } Integral = 3

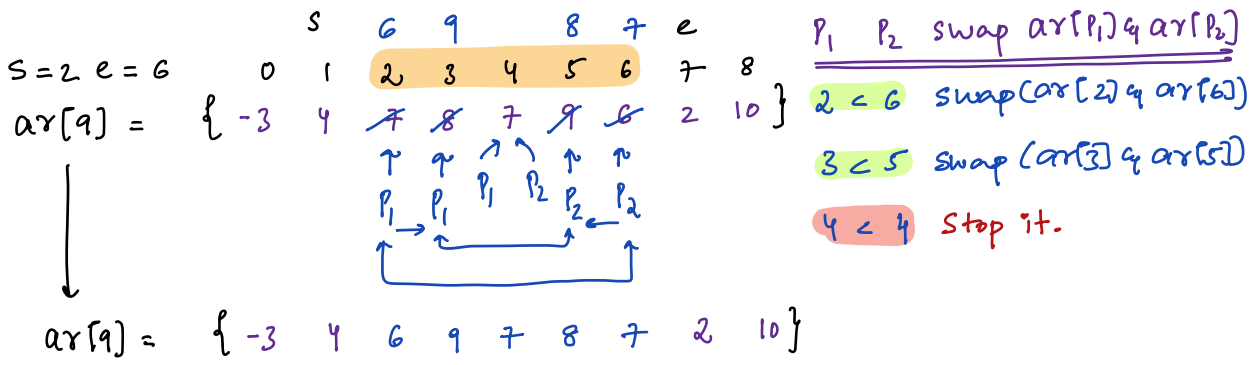
$\log_2^{20}$ = $2^4 = 16$, $2^5 = 32$ → $2^{4.n} = 20$ } Integral = 4

28) Given ar[N] elements & index s & e 10:20pm

Reverse ar[] from index [s e]

Note1: Need to modify given ar[] itself, Extra Array Not Allowed.

Eg1:



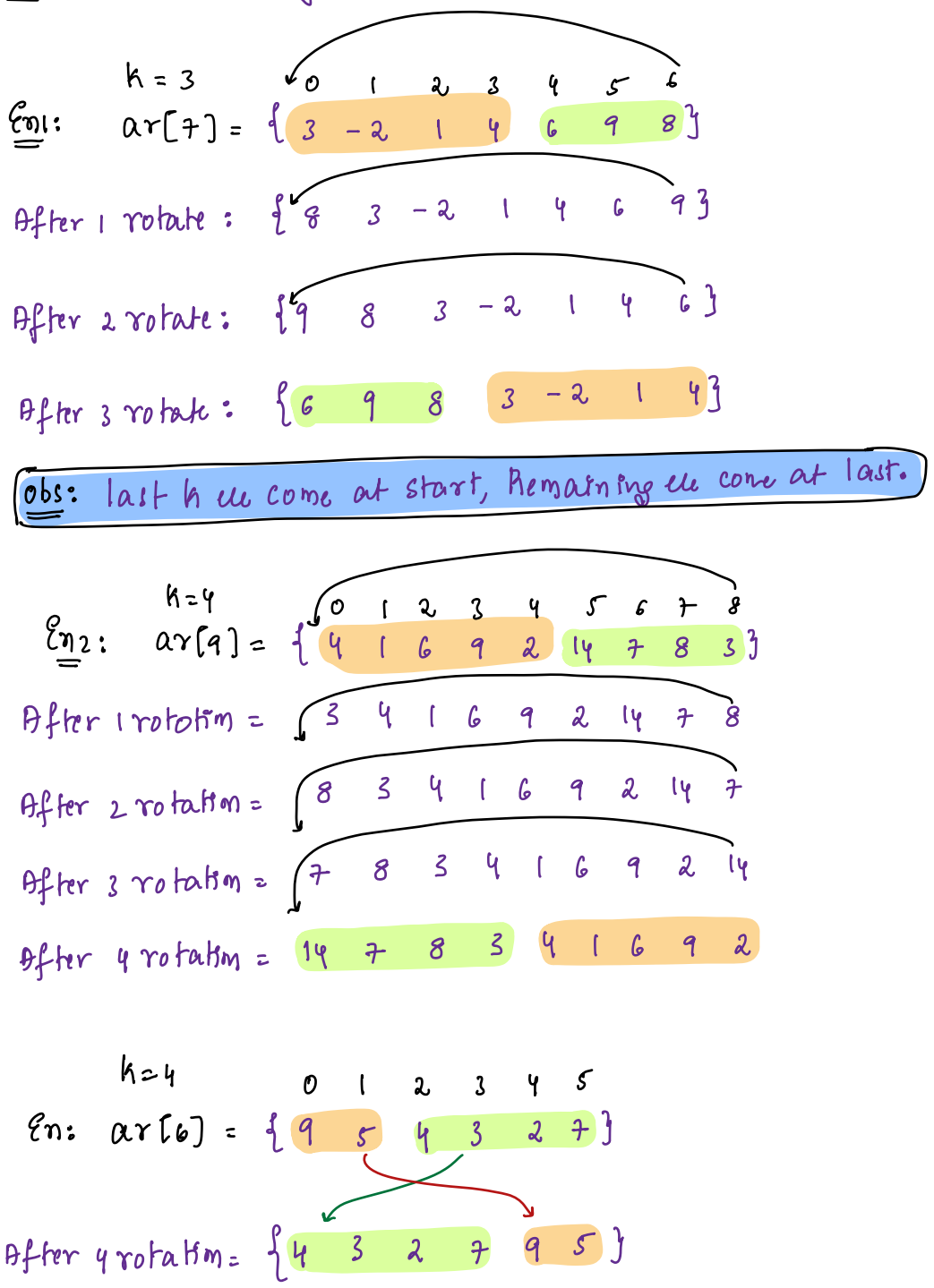
P_1	P_2	swap ar[P_1] & ar[P_2]
2 < 6		swap(ar[2] & ar[6])
3 < 5		swap (ar[3] & ar[5])
4 < 4		Stop it.

```
void Revenrange (int ar[], int s, int e){
    int p1 = s, p2 = e;
    while( p1 < p2 ){
        //swap ar[p1] & ar[p2]
        int tmp = ar[p1]
        ar[p1] = ar[p2]
        ar[p2] = tmp
        p1++; p2--;
    }
}
```

Rotate : 10:40pm

30) Given ar[N] & K : Rotate array from last to first k times

Note: Need to modify given ar[] itself, Extra Array Not Allowed.



Eg: 4:

k = 5

ar[13] = { a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 | a_8 a_9 a_{10} a_{11} a_{12} }

1. reverse ar[] { a_{12} a_{11} a_{10} a_9 a_8 | a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0 }

2. reverse first 5 ele

3. reverse remaining ele

a_8 a_9 a_{10} a_{11} a_{12} | a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7

After rotations { a_8 a_9 a_{10} a_{11} a_{12} | a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 }

Steps: Rotate ar[N] by k times:

1. Reverse entire arr[] ⟶ reverse ar[] from [0.. N-1]
2. Reverse first k elements → reverse ar[] from [0.. k-1]
2. Reverse remaing elements → reverse ar[] from [k.. N-1]

```
void ReverseRange(int ar[], int s, int e){
    int p1 = s, p2 = e;
    while( p1 < p2 ){
        //swap ar[p1] & ar[p2]
        int tmp = ar[p1]
        ar[p1] = ar[p2]
        ar[p2] = tmp
        p1++; p2--;
    }
}
```

```
void rotate(int ar[], int k){
    int n = ar.length;
    k = k%n // add this line, it will work
    ReverseRange(ar, 0, n-1)
    ReverseRange(ar, 0, k-1)
    ReverseRange(ar, k, n-1)
}
```

	0	1	2	3
Eg: ar[4] =	1	2	3	4

k = 6

N=4 k=6

ReverseRange(ar, 0, 3)

ReverseRange(ar, 0, 5) Error:

Eg: arr[4] =

0	1	2	3
1	2	3	4

k = 6

rotate 0 = 1 2 3 4

rotate 1 = 4 1 2 3

rotate 2 = 3 4 1 2

rotate 3 = 2 3 4 1

rotate 4 = 1 2 3 4

rotate 5 = 4 1 2 3

rotate 6 = 3 4 1 2

rotate 7 = 2 3 4 1

rotate 8 = 1 2 3 4

After 4 rotations

We got original arr[]

0 → 4 → 8 → 12 → ...

1 → 5 → 9 → 13 → ....

2 → 6 → 10 → 14 → ...

3 → 7 → 11 → 15 → ...

k = k % N = Same

20 % 4 = 0

17 % 4 = 1

26 % 4 = 2

30 % 4 = 2

35 % 4 = 3

4 is arr.length;

Announcement

a) Time Management: To Solve Assignments?

Motivation is lacking

Scared...

Schedule:

Mon | Tue 9pm | Wed | Thu 9pm | Fri | Sat 9pm | Sun

Mon/Wed/Friday: 9pm - 11:30pm For Scaler Assignments?

Trying: 1 Week: Pratice Sessions/Scaler / Optional.

Mon 9pm - 11pm | Wed 9pm - 11pm | Fri 9pm - 11pm

What are these sessions?

a) Open previous class assignment & solve them.
b) Open previous class homeworks & solve them.
c) Once above 2 are done, open backlogs
d) No Teaching from my side.

Audio/Video/TA:

Doubt/unmute me/Share your Screen/Debug.

a) Code ⟶ Issues
   a) Will make you see issue
   b) Give hints to solve
   c) Give hints for Edge Cases
b) Concept, last class:
   a) Watch recording.