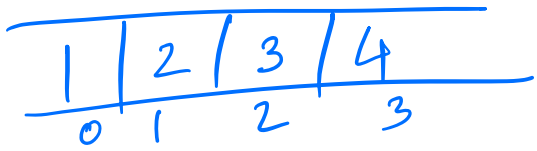
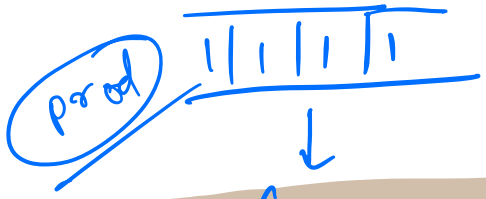




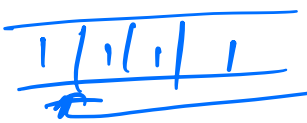
temp = 1

```
for( i = 0 ; i < n ; i++ )
{
```

```
prod[i] = temp
temp *= arr[i]
```

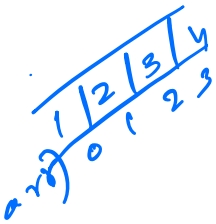
temp has mul value until i-1

i = 0



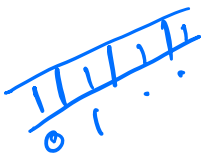
prod[0] = 1

temp = temp * 1 = 1 * 1 = 1



i = 1) prod[1] = temp = 1

temp = 1 * arr[1] = 1 * 2 = 2

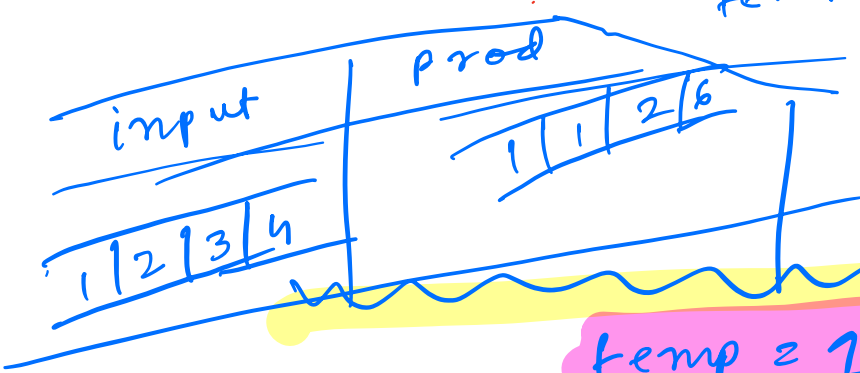


i = 2) prod[2] = temp = 2

temp = temp * arr[2] = 2 * 3 = 6

i = 3) prod[3] = temp = 6

temp = temp * arr[3] = 6 * 4 = 24



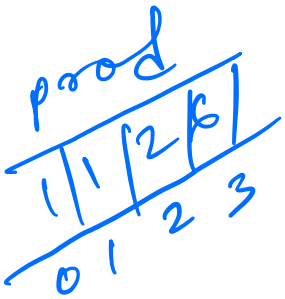
temp[i-1] is in prod[i]

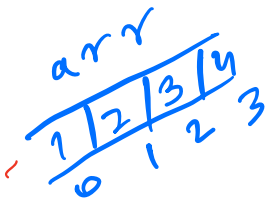
temp = 1

```
for( i = n-1 ; i >= 0 ; i-- )
{
```

prod[i] = prod[i] * temp

temp = temp * arr[i]





i = 3) prod[3] = prod[3] * 1

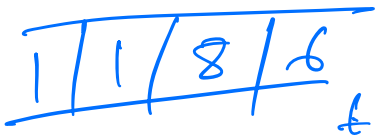
= 6 * 1 = 6

temp = 1 * 4 = 4

i = 2) prod[2] = prod[2] * 4

= 2 * 4

= 8



temp = temp * arr[2]

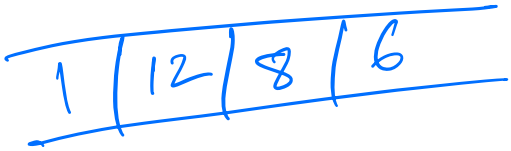
= 4 * 3 = 12

i = 1) prod[1] = prod[1] * temp

= 1 * 12 = 12

~~temp = 12 * arr[1]~~

= 12 * 2 = 24



i = 0) prod[0] = prod[0] * temp

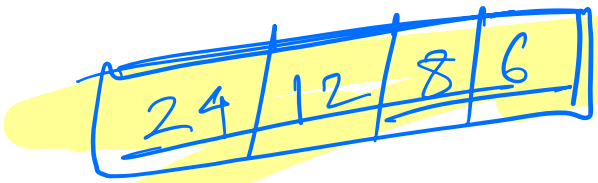
= 1 * 24.

= 24

temp = 24 * arr[0]

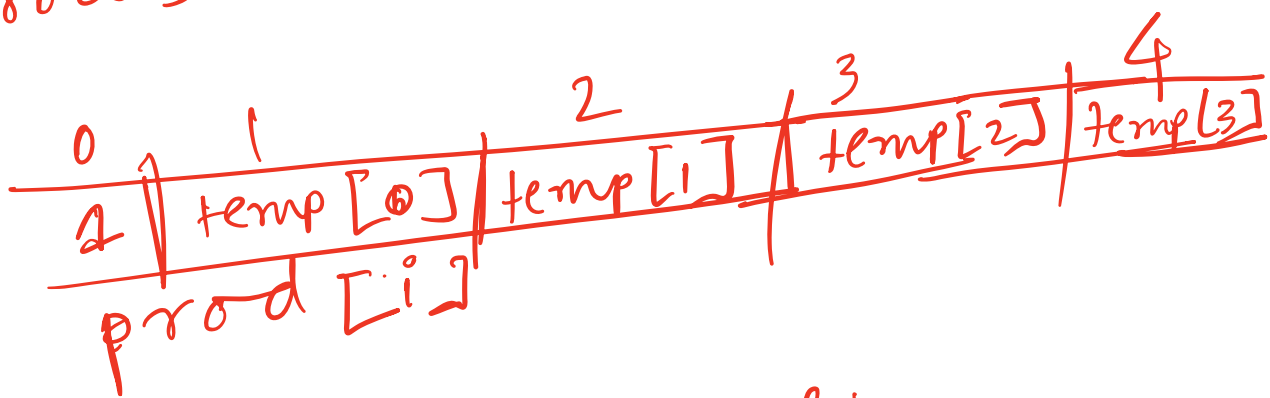
= 24 * 1

= 24



Summary

On the first pass, temp grows towards the right



Thus at any point of time,

prod[i] = temp[i-1]

↳ temp grows by multiplying numbers

Hence, while coming back to calculate prod[i] we need a similar temp growing from the opposite direction.

prod[i] = temp_previous[i-1] * arr[i+1]

in next step

=> prod[i] = prod[i] * temp

temp = temp * arr[i]

this is growth

And also post update to the next (0/1)