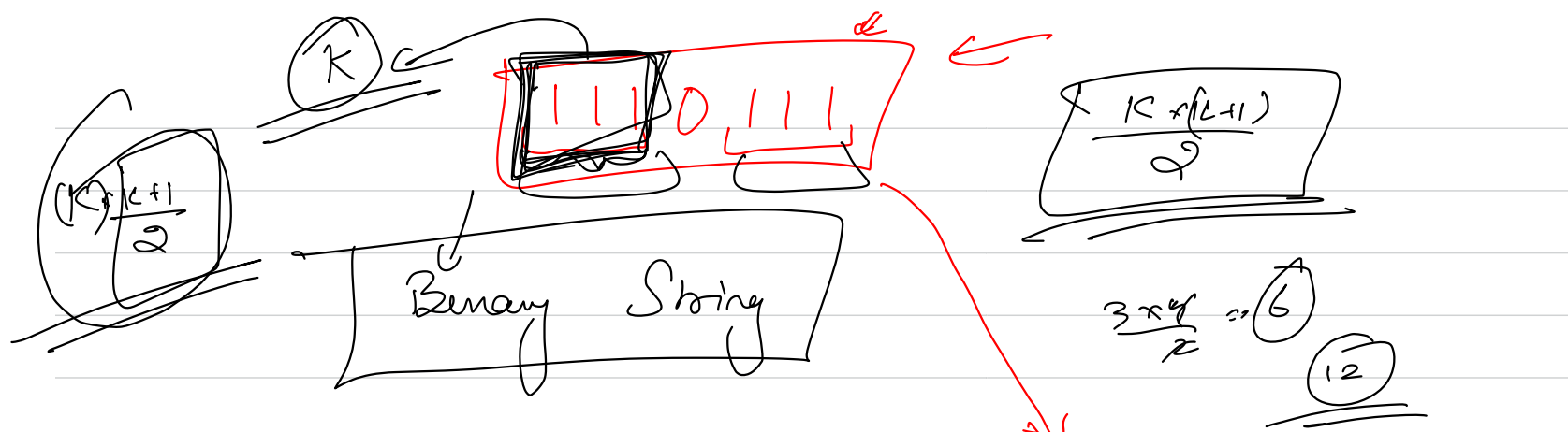abac ab a

a, b

anyhow if we manage to keep the set of allowed chars
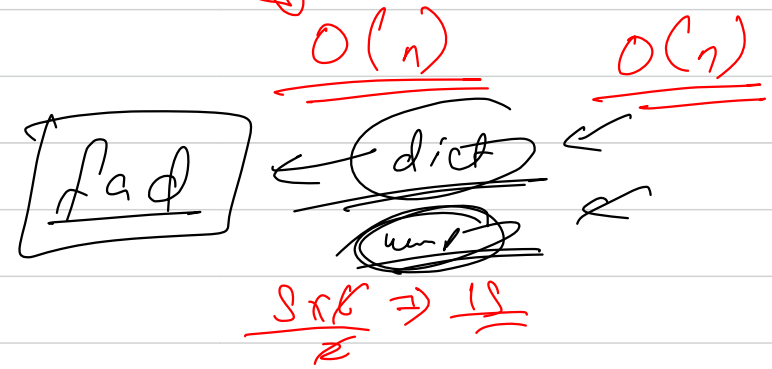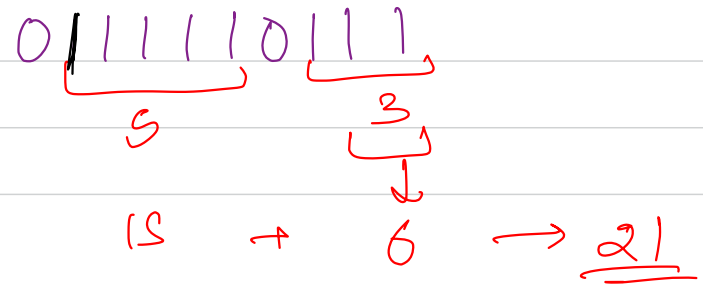
allowed chr ⟶ 1
not-allowed chr ⟶ 0

$\left(\dfrac{K \times K+1}{2}\right)$

K

$1\,1\,1\,0\,|\,1\,1\,1$ ←

$\dfrac{K \times (K+1)}{2}$

Binary String

$\dfrac{3 \times 4}{2} = 6$

12

$O(n)$ $O(n)$

fad ← $dict$ ←

um

$\dfrac{3 \times K}{2} \Rightarrow 15$

sadfaasdda

$0\,|\,1\,1\,1\,1\,0\,|\,1\,1$

5 3

$15 \quad + \quad 6 \quad \longrightarrow \quad 21$

ℓ

$\boxed{\ell, r}$

$\boxed{1\ 1\ 1\ 1}\ 0\ 1\ 1\ 1\ 0\ 1\ 1$

$\underbrace{\qquad}$

$1, r$

$\uparrow$

$2\ ph$

custom objects → i = 10 integer

i = 5.5 float

i = person

i = place
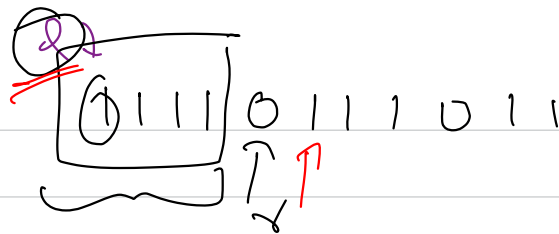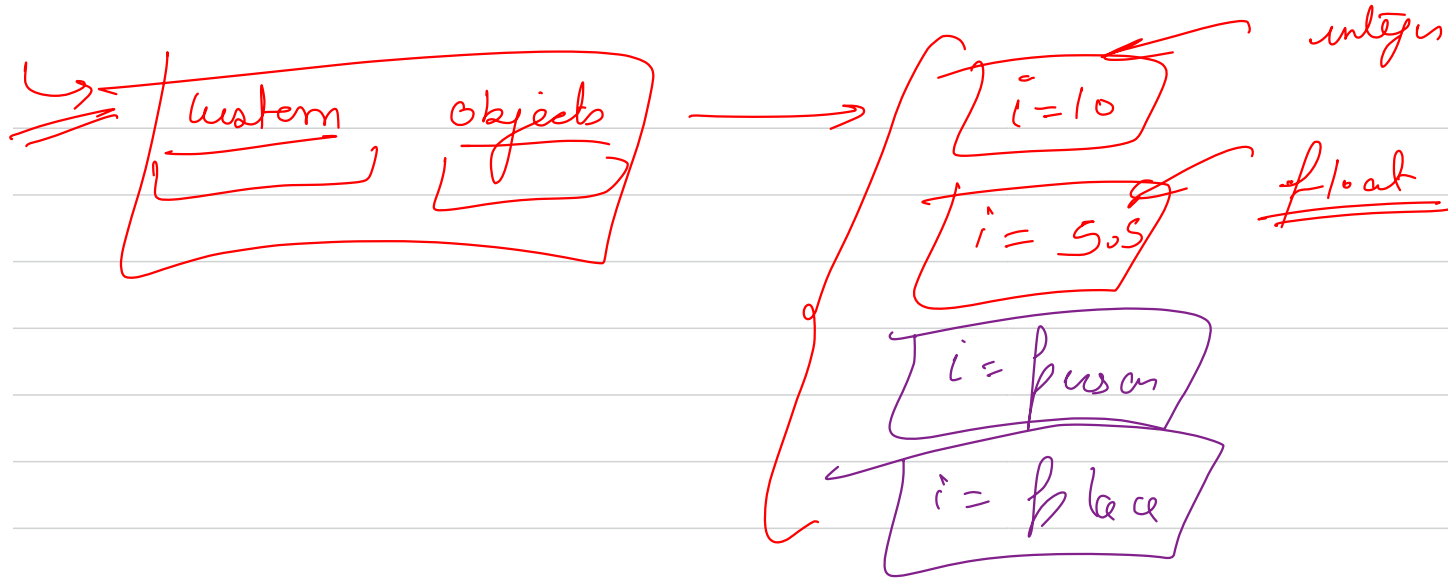
$\rightarrow$ Object Oriented programming **SPS**

- $\rightarrow$ functional $\rightarrow$ perl, JS etc
- $\rightarrow$ procedural
- $\rightarrow$ OOP $\rightarrow$ classes / objects
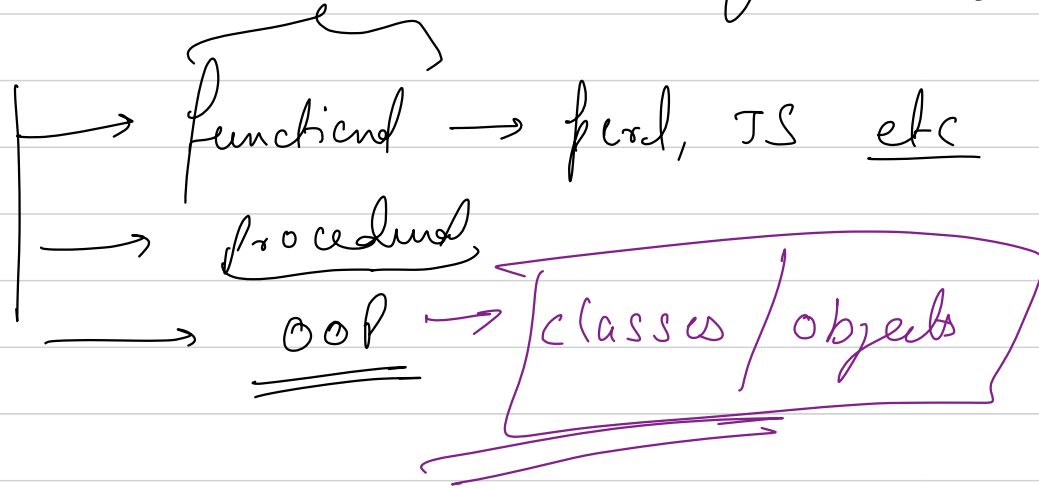
Flipkart
Amazon

products

$\rightarrow$ name
$\rightarrow$ desc
$\rightarrow$ price
$\rightarrow$ discon
:
:

class

OOP

classes
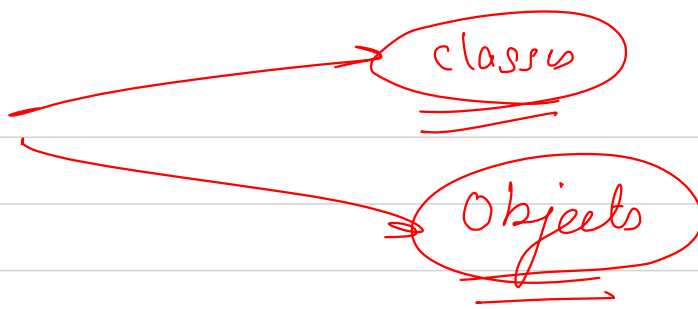
Objects

Real World entities uses classes and objects

Dict → { key - val
                ↑
                ⋮
         }

Terminology

a) **Class** → It is a user defined prototype/blueprint for a real world object that defines properties of the object

b) Object → Instance → An individual real world entity that is based on the blueprint of some class.
↳ unique instance of a class

Product → Name → string — Properti
→ [discount] → dec ↓
→ original price → int — date member
→ close → string

Real price

class

object → Sch— 101 headphe
→ 30%
→ 1000
Loremipsum

SDE
Design

functional — update discount
→ real price

methods

SDE
SWE
SE

Python → Constructor → special function which is used to initialize a brand new object from a class

Product class

name: Macbook Pro
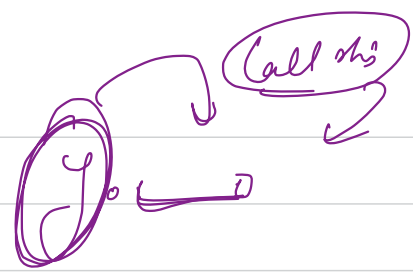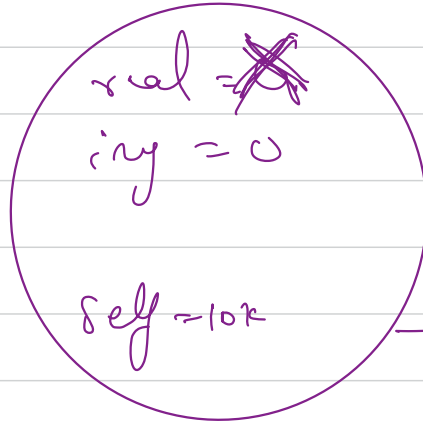
price → 300000

discount → 2.5

desc → " ———— "

self → 10^{10}

Java → this
C++ → this
Python → self

Obj

→ 10K → address

real = ~~0~~

imy = 0

self = 10k $\longrightarrow$ ~~y~~

self.real = X

$\rightarrow$ $O(O^\downarrow N^\downarrow)$ $\rightarrow$ optimum

$\rightarrow$ Prefix Sums / Cumulative Sum

You have an array/list $\underline{arr}$,

$$f(i) = \sum_{k=0}^{i} arr[k]$$

prefix sum
till the ith index

$$0 \quad 1 \quad 2 \quad 3 \quad 4$$

$$1, \; 2, \; 3, \; 4, \; 5 \quad \rightarrow 9 \qquad r = 3$$
$$l = 1$$

$$0 \quad 1, \; 3, \; 6, \; 10, \; 15$$
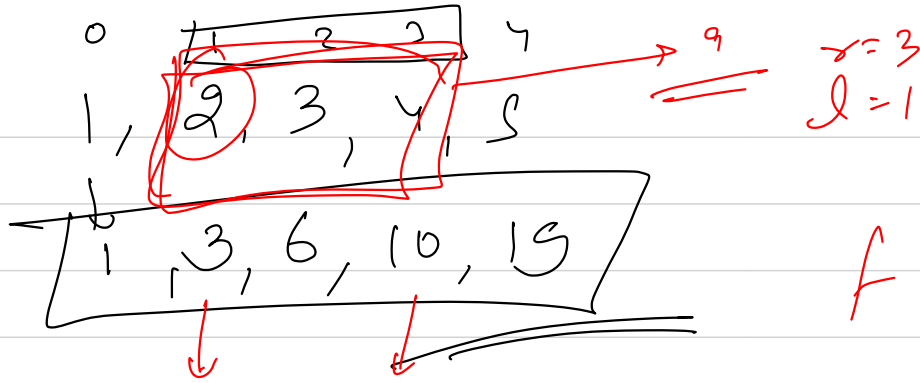
$$f(i) \rightarrow \text{prefix sum till}$$
$$i$$

$$Sum(l, r) = f(r) - f(l) + arr[l]$$

$$10 - 3 + 2$$

$$\emptyset \rightarrow O(1) \qquad \rightarrow \quad 7 + 2$$
$$\rightarrow 9$$

$$O(\emptyset)$$

Q → You have a list of size n, of all zeroes.
You will get q queries & in each query
you will get 2 number l, r. You need to
increment all the element from l to r by 1.
Print the final list after processing all queries

$$[0, 0, 0, 0, 0]$$

$n \leq 10^7$ $q \leq 10^5$

(q=3)

$$\begin{array}{cc} l & r \\ 1, & 3 \\ 2, & 2 \\ 3, & 4 \end{array}$$

$$\overset{0 \quad 1 \quad 2 \quad 3}{\longmapsto \quad [0, 1, 2, 2, 1]} \quad \text{ans}$$

$\longrightarrow$

$[0, 0, 0, 0, 0]$

$\phi \leq 10^5$

$n \leq 10^7$

$\phi \sim$

l, r

$O(\phi \cdot w)$

read me q curive
all me q curive

$O(w)$ atчаст

Start from l^{th} idx
till r^{th} indx incrent
values

$< \phi \cdot w$

later process all
the queries at
once

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

$[\ 1,3\ ]$
$2,2$
$3,4$
$1,4$

$arr[l] += 1$

$arr[r+1] -= 1$

$l \rightarrow$ is my
first offected
indut

$r+1 \rightarrow$ is my
first unaffected
inden to my right
of l

$\rightarrow$ prefix su

$0, 2, 3, 3, 2$

prefix sur

$O(\phi + m)$

$[\ 0, 1, 2, 2,]$
$\quad 2 \ 3 \ 3 \ 2$

Difference
arras
trick

$arr[l]\ +\ =1$

$arr\ [r+1]\ -=)$

prefer Sm

$r+1 < n$