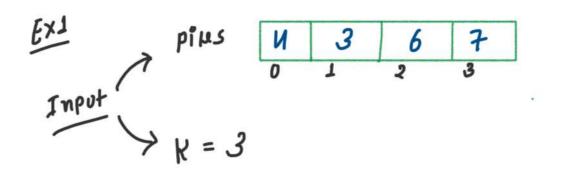
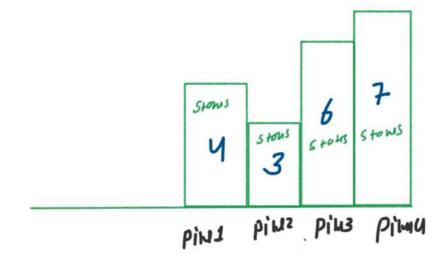
26/12/2023

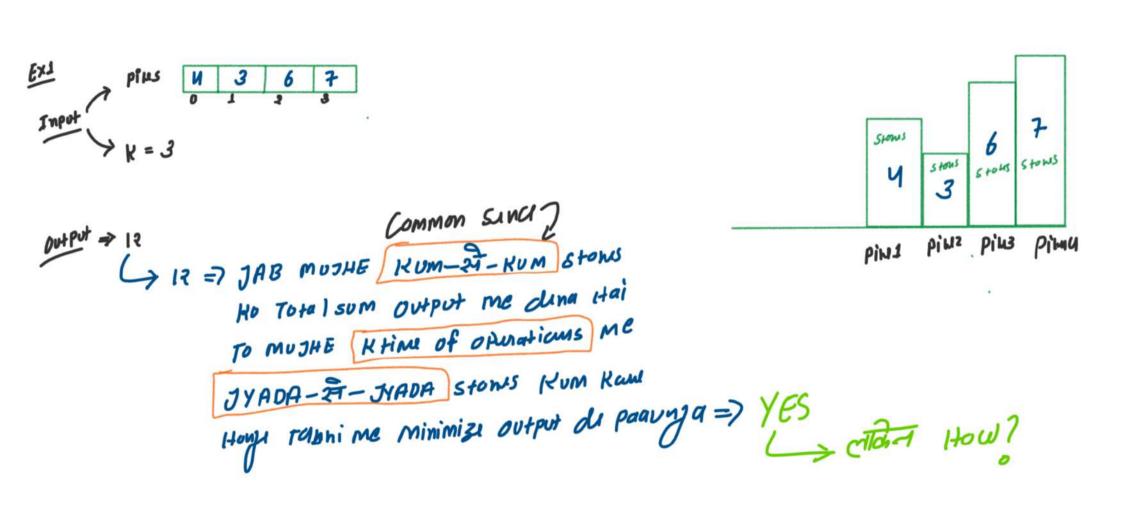
## HEAP CLASS - 4



1. Remove Stones to Minimize the Total (Leetcode-1962)







STEP1 Find max pill from piles arrage

STEP2 pur form the operation to minimize the stones pile

N = 3

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile

| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation to minimize the stones pile
| STEP2 | Pur form the operation the STEP3 VIETUM TOTAL OF MIN STOWS PIW FORM HUAP

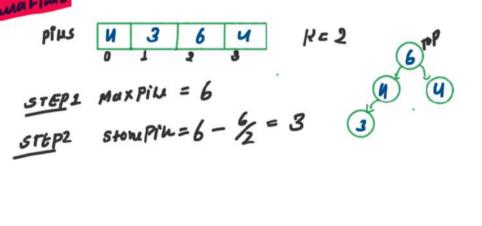
MAX PIN FIND Kanne Ruliff Hum MAX Huap Ha USA

Kannya Kyunki Time cumplu kity O(1) Hoti Hai

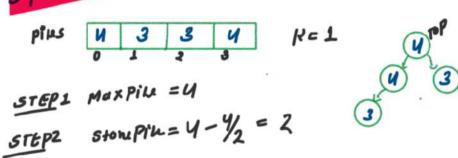
## DRY RUN

## Opma+1

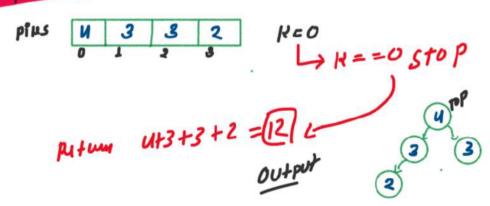
### n D. Marima



### puatim3



## o pratimy



```
. .
class Solution {
    int minStoneSum(vector<int>& piles, int k) {
       priority_queue<int> maxHeap;
        for(int pile = 0; pile < piles.size(); pile++){
            int stonesPile = piles[pile];
            maxHeap.push(stonesPile):
       while(k--){
            int maxStonesPile = maxHeap.top();
            maxHeap.pop():
            maxStonesPile = maxStonesPile - floor(maxStonesPile/2);
            maxHeap.push(maxStonesPile);
            int top = maxHeap.top();
            sum += top;
            maxHeap.pop();
```

```
> CREATE MAX HEAP WITH T.C.
```

Max Huap takes
Span U(N)

N = Total Elimints K = 0 pmations

Minimize the stoms PiW With T.C.

=) O(K\*109 N)

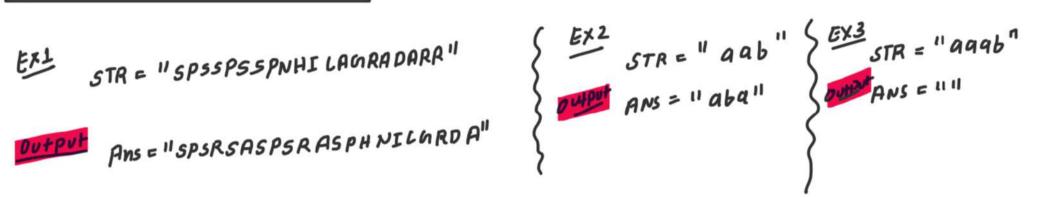
Get total sum of minimize stones pin with F.C. => O(N\*10) P)

= 0(N × 10/N) + 0(K × 10/N) = 0(N × 10/N)



### 2. Reorganize String (Leetcode-767)





#### Problem Statement:

Given a string s, rearrange the characters of s so that any two adjacent characters are not the same.

Return any possible rearrangement of s or return "" if not possible.

STR = "SPSSPSSPNHI LAGRADARR"

store frequing of all character in an Assag .

Count Fouquet

P = 3

N = 1

H =1

I =1

L =1

A = 3

6 =1

R = 3 D = 1

Cough maxtuap to push all charactus frequency where frequency >0

STEP3 Re-organize the storing and verture String

```
STEPL
                                                                                                                               112 p
                                                                                                                  81 Q
                                                                                                                               113 q
                                                                                                                               114 r
                    STR = "SPSSPSSPNHI LAGRADARA"
                                                                                              ASCI
                                                                                                                               115 s
                                                                                                           68 D
                                                                                                                  84 T
                                                                                                                               116 t
                                                                                              LODE
                                                                                                           69 E
                                                                                                                               117 u
                                                                                                           70 F
                                                                                                                        102 f
                                                                                                                               118 v
                                                                                                                  87 W
                                                                                                                               119 w
                                                 ASCII CODE OF S = 115
                                                                                                                        103 g
                                                                                                           72 H
                                                                                                                        104 h
                                                                                                                              120 x
                                                                                                          73 I
74 J
                                                                                                                  89 Y
                                                                                                                       105 i
                                                                                                                              121 y
                                                ASCA CODE OF a = 97
                                                                                                                 90 Z
                                                                                                                              122 z
                                                                                                                        106 j
            Count Forguet
                                                                                                           75 K
                                                  STORE FREquely of 5 = frum ['3-19']++
=) frum [115-97]++
                                                                                                                        107 k
                                                                                                                        108 1
                                                                                                          77 M
78 N
79 O
                                                                                                                        109 m
                                                                                                                        110 n
PII chapacturs
are in Lown
                                                                                                                       111 o
                                      frequery [26] = {0}
                                                                                 => FUICIBJ++
                                                                0
                                                                      0
                                                                                             0
                                                                                                  0
                                                                                                               0
                                                                                                                               3
                                                                                                                         0
                                                           3
                                                                                             9
                                                                       5
                                                                                                   10
                                                                                                        11
                          =1
                                                                                                     ...
                                                 0
                                                         0
                                                                               0
                                         6
                                                                 0
                                                                        0
                                                                                     0
                                                                                           0
                          =3
                                                                                                         int frequency[26] = {0};
                                        16
                                                                       22
                                                                               23
                                                19
                                                        20
                                                                 21
                                                                                      24
                                                                                           25
                                                                                                           frequency[s[i]-'a']++;
                          =3
```

### STEP 2

Maxtuap ka Each
Nod Kya
Store Hauja?
NOD 6

```
Max Huap
S = 6
P = 3
N = 1
I = 1
I = 3
G = 3
P = 1
```

```
class Info {
    public:
        char ch;
        int count;

        Info(char ch, int count){
        this->ch = ch;
        this->count = count;
    }
};

class Compare {
    public:
        bool operator()(Info* first, Info* second){
            return first->count < second->count;
    }
};
```

```
// Step 2: create max heap to push all characters frequency where frequency[i] > 8
priority_queue<Info*, vector<Info*>, Compare> maxHeap;
for(int i=0; i<26; i++){
   if(frequency[i]>0){
        Info* tempNode = new Info(i+'a', frequency[i]);
        maxHeap.push(tempNode);
   }
}
```

## STEP3

- (I) Get Two characturs from maxtuap and POP - JINKI MAX free grung Ho
- D push Both enanactus into a New string Ares and dicriment the count of bolk.
- Again push both charactur jnto MaxHuap
  where out >0

Repeat All stup jab tak max Hap-sizel) >1

MaxHuap

(II) Last Charactum 120 Alaj SI Handu Kan lengs

Last Charactum 12

[Finst]

(SLEUND)

(SECOND)

(SECOND)

(SECOND)

(SECOND)

(MAX HEAP

$$S = 6$$

$$P = 3$$

$$N = 1$$

$$H = 1$$

$$I = 1$$

$$L = 1$$

$$S > 70$$

$$S > 70$$

$$S > 70$$

$$MHo push ( Finst)$$
(Max Heap

$$N = 3$$

$$N = 1$$

$$H = 1$$

$$L = 1$$

$$L = 1$$

$$L = 1$$

$$R = 3$$

$$D = 1$$

Sigl = 10

[Finst]

(Size and A = 3)

(Max Huap

Max Huap

$$S = 5$$
 $P = 2$ 
 $N = 1$ 
 $N = 1$ 

Itumtion 3

[Finst]

(SLEUND)

(SECOND)

(SECOND)

(SECOND)

(SECOND)

(MAX HEAP

MAX HEAP

$$S = U$$

$$P = 2$$

$$N = 1$$

$$H = 1$$

$$T = 1$$

$$3 > 0$$

$$(UMt > 0$$

$$3 > 0$$

$$(UMt > 0$$

$$3 > 0$$

$$MHo push (Second)$$

MHo push (Second)

Size = 0

Ituationu

Itumations

[Finst]

(Second)

(E) 
$$S = 2$$
 and  $A = 2$ 

(D)  $ASN = [SPSA]SRSPSA$ 
 $S = 2$ 
 $P = 1$ 
 $S = 1$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S = 1$ 
 $S = 2$ 
 $S = 1$ 
 $S =$ 

MaxHuap Ituation 6 S M4 count = 0 } A M4 count = 1 MHOPUSH (First)X MHOPUSH (Second)

Si 21 = 10

$$\begin{array}{ll} \text{(SLCUMd)} \\ \text{(D)} & P = 1 & \text{and} & N = 1 \\ \\ \text{(D)} & P = 1 & \text{and} & N = 1 \\ \\ P & \text{(N)} & P & \text{(SLCUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(II)} \\ P & \text{(SICUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(III)} \\ P & \text{(SICUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(III)} \\ P & \text{(SICUMd)} \\ P & \text{(SICUMd)} \\ P & \text{(III)} \\ P & \text{(SICUMd)} \\ P & \text{(III)} \\ P & \text{(SICUMd)} \\ P & \text{(III)} \\ P & \text{(IIII)} \\ P & \text{(III)} \\ P & \text{(III)} \\ P & \text{(III)} \\ P & \text{(III)}$$

siz1=7

Ituatiu8

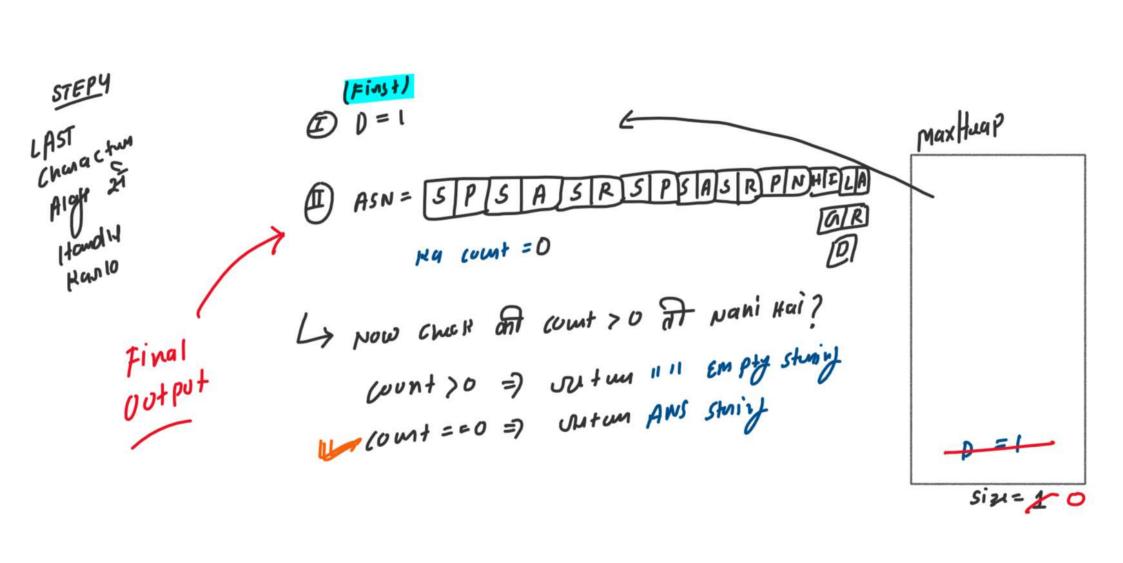
MaxHuap Ituation 9 L Ma count = 0 } A Ma count = 0

siz= 5

Ituation 10 GIR ON MY COUNT = 0 } R MY COUNT = 0 MHopush(second)X

MHopush(second)X

Si 21 = 3



max Huap

last character du Bu Play se Hardy

- (First = 9 court = 2

max Huap

Count 70 (170)

(Re-onjanization of input

(Re-onjanization of input

string is Impossible)

```
...
class Info
        char ch:
        int count;
class Compare
        bool operator()(Info* first, Info* second){
class Solution {
    string reorganizeString(string s) {
                                              S.C. = O(1) STEP 1
        int frequency[26] = {0};
            frequency[s[i]- a ]++;
               Info* tempNode = new Info(i+'a', frequency[i]);
maxHeap.push(tempNode);

5.(. = O(1))

3: reorganize the string

5.C. = O(N) STEP3
```

#### STEP 3

. .

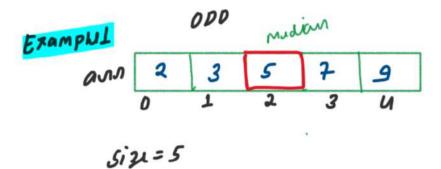
```
string ans = ";
while(maxHeap.size() > 1){
    ans.push_back(first->ch):
    // Agar dono ke count > 0 hat to fir se maxHeap me unbe push kardo \ref{first->count} > 0){
   abhi maxHeap me last character bacha huas hai
usse bhi push krdo ans me
    if(last->count > 0){
```

```
Time complexity
           T.C. = O(N), when N is Numbers of different changets of string
       FUNLOOP KI TIC. = O(N) } > O(N * 10) N)
MAXHLAP = O(19) N)
          maxtuap = OLLyN) > O(N * 19) N)
O UMAII T.C. => D (N * 10 ) Space compM xity

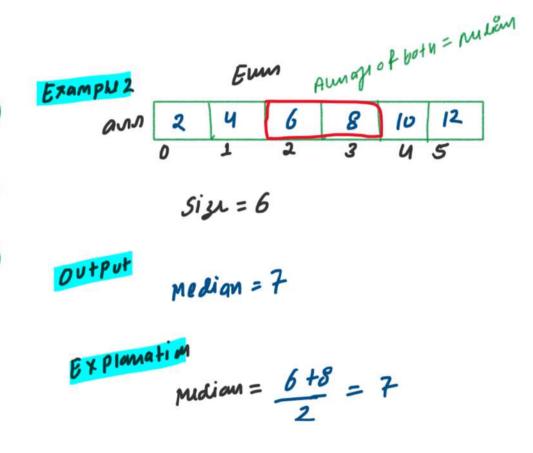
O (N), AND STAIN FORMS SPACE S PACE O(N)
```



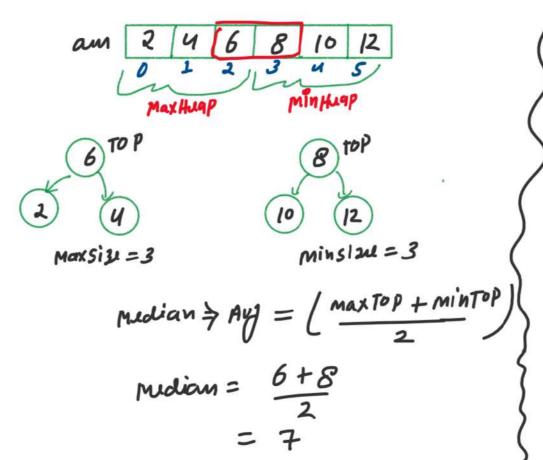
### 4. Median in a Stream (Coding Ninjas)

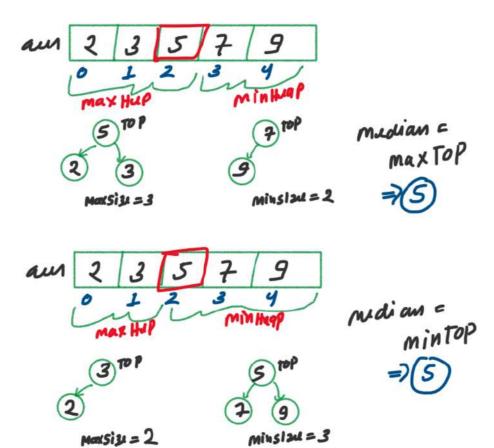


# Byplanatiam



Note: median Find Kann Ke ligh suius sont Honi Chigh





me liam = 
$$\frac{\text{max Top + min fop}}{2}$$

when  $\text{max six} = = \text{min six}$ 

Diff = 0

Insurtion of a summit kante Time Two point Always Yand Rakne Hope

point 1

Absolute diff." of max Huap and min Hun = 0 ya 1 Hona chajye

pointz

maxtual + mintual => SORTED Somes Honi chaigh with

Cases of Insunting Elevent

(I) maxsize = minsize

CANI FRANT > Molions

Limit > min Huap, posh ( Ehmt)

Eunt 2 muliam

L) max Huap. push( Eunt)

maxsize = = minsize+1

Elmt > min Huap. push (Elmt)

Eunt 2 modions

De max Top min Huap

De sunt push

max Huap

minsize == maxsize +1

(I) ELM+ > mudium

| Down max Huap

(I) ELM+ > mudium

| Down max Huap

(II) ELM+ | Push min Huap

Ly max Huyp. push (Elut)

8 4 2 3 15 2 3 4 5 6 SORT 2 3 4 8 10 12 15 Ly we are finding median witnout conting ]



ang	12	10	8	4	2	3	15
	0	1	2	3	4	5	6

initionly

median = 0

Max

WIN

NO=12

Itmat-D

Size = 0

Si 31 = 0

if (maxsize = minsize)

NO > median

12 > 0

min Ituap. posh( 12)

Mediam = |2| max |2| min |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3| |3|

median = g max 
$$u = 0$$
 min  $u = 0$ 

$$f(maxsize = minsize)$$

$$max = minsize$$

$$min = 0$$

$$max = 0$$

mediam = 6 max 
$$234$$
 min  $81012$   $N0 = 15$ 

maxize = minsize )

mindiam =  $\frac{d+8}{2}$ 

minhuap pus h(15)

No > mediam

15 7 6

Point 2 0/1

Point 2 SORTEDS.

23481012

median = 
$$\max_{Size = 3} 234$$
  $\max_{Size = 3} 81012$ 

if (minsize =  $\max_{Size = 3} 111$ )

 $\max_{Size = 3} 111$ 
 $\max_{Size = 3} 112$ 
 $\min_{Size = 3} 112$ 

Point 2 0/1

point 2 SORTED So

2348 10 12

```
// PROBLEM 4: Median in a Stream (CodingNinjas)

#include<iostream>
#include<queue>
#includevector>
using namespace std;

void solveForMedian(int &median, priority_queue<int> &maxHeap, priority_queue<int, vector<int>, greater<int>> &minHeap, int element){
    int main(){
        int arr[] = {12, 10, 8, 4, 2, 3, 15};
        int median = 0;
        priority_queue<int> maxHeap;
        priority_queue<int> maxHeap;
        priority_queue<int, vector<int>, greater<int>> minHeap;

        for(int i=0; i<n; i++){
            int element = arr[i];
            solveForMedian(median, maxHeap, minHeap, element);
            cout << "Median foud: " << median << endl;
        }
        return 0;
}
</pre>
```

```
...
void solveForMedian(int &median, priority_queue<int> &maxHeap,
priority queue<int, vector<int>, greater<int>>> &minHeap, int
    if(maxHeap.size() == minHeap.size()){
       else
            minHeap.push(element);
```

N = Total Eliments of oursay

FOR LOOP KIT.C.

OLN) Ollojn) OWNAIL TOC. =)

O(N \* 199 N)

SPACE COMPLEXITY

SPACE O(N)