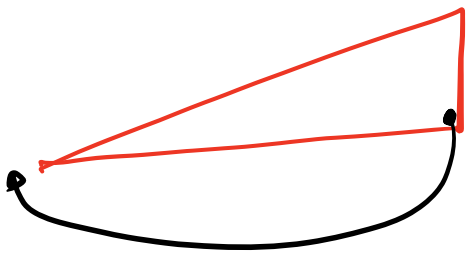
Q Given an array which is formed by rotating a sorted Array K times!

Given X. find the index of it in this Array.

DISTINCT



A: [1 2 3 4 5]

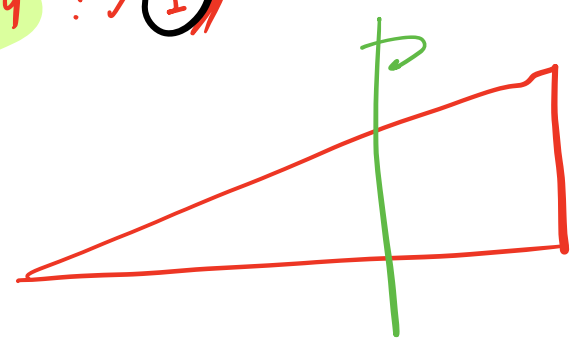
K=1 5 1 2 3 4

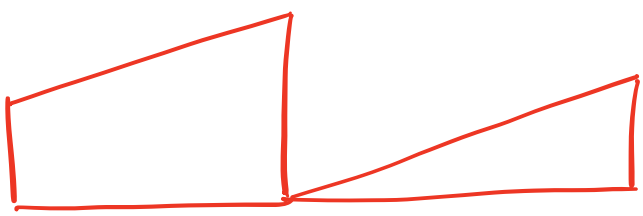
K=2 4 5 1 2 3

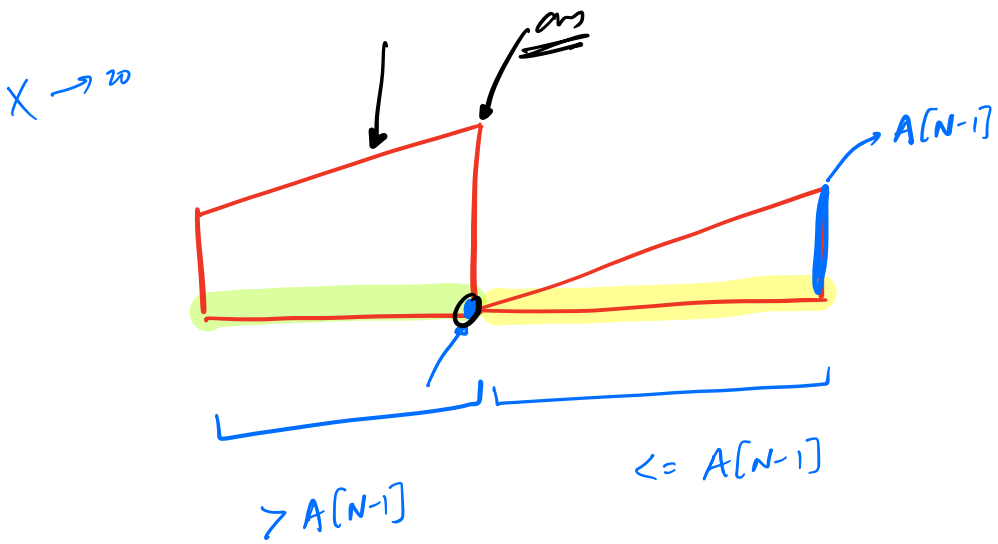
K=3 3 4 5 1 2

NOT GIVEN K

X → 4 : 1





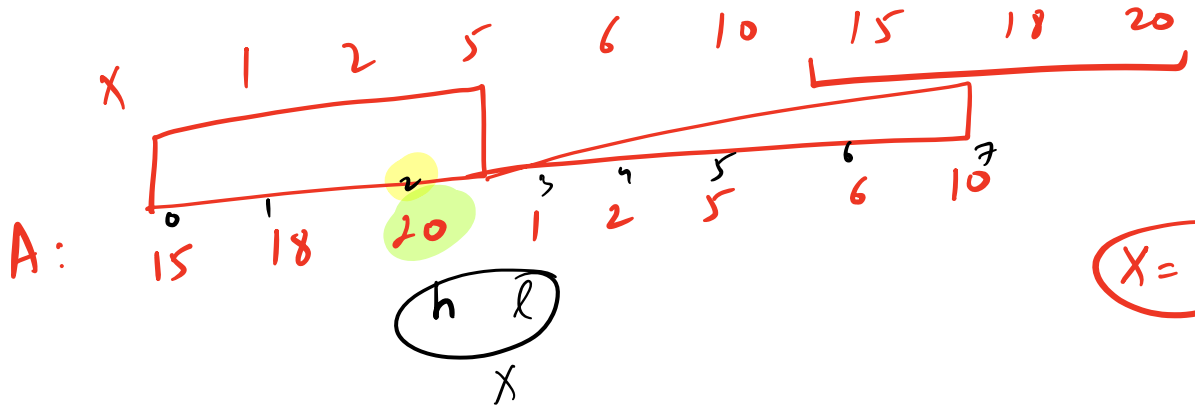



```
l = 0,  h = N-1 ,  ans = -1;
while( l <= h) {
    m = (l+h)/2;
    if ( A[m] > A[N-1]) {
        ans = m;
        l = m+1;

    }
    else {
        h = m-1;

    }
}
if(x > A[N-1])
    BS( A[], 0, ans, X)
else
    BS( A[], ans+1, N-1, X)
```

SS → N

TC = O(log N)

SC = O(1)

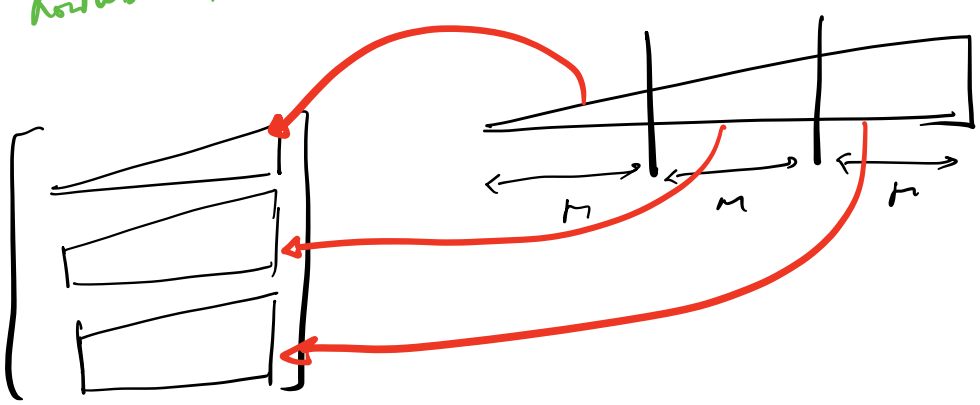


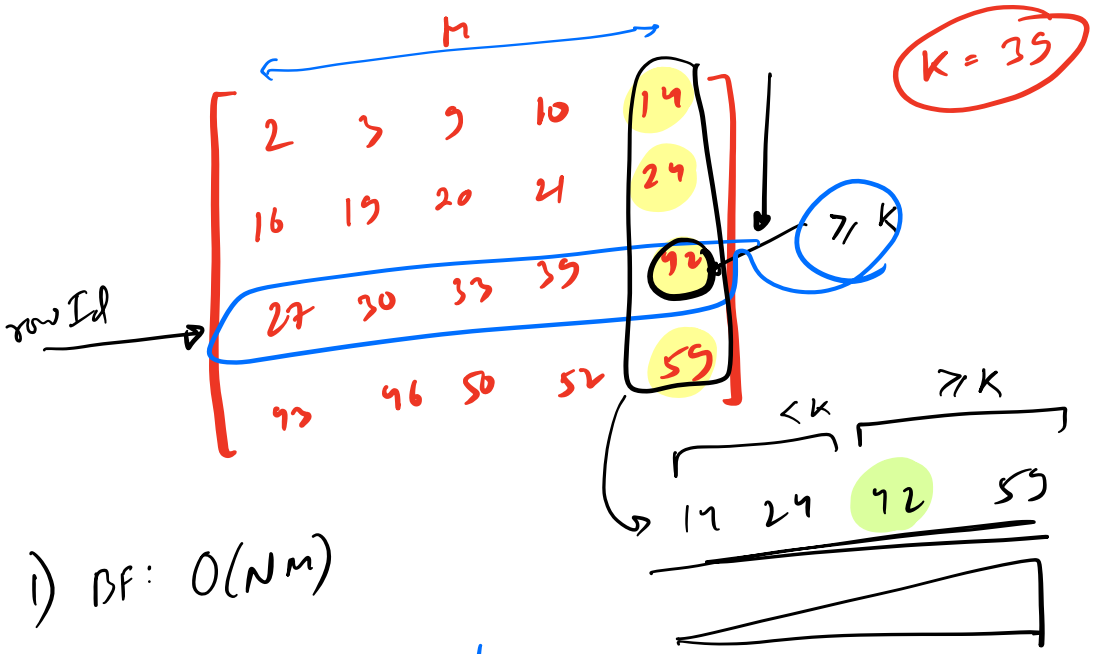
any:
~~1~~
X
2

Q Given a 2D sorted matrix. Search for K.

DISTINCT!

↳ Rowwise & Colwise sorted





I) BF: $O(NM)$

II) BS on every row!

$1 \rightarrow \log(M)$

$N \rightarrow O(N \log(M))$

III) find the first row s.t. the last elmt of it $>= K$

↳ BS in it!

$O(N + \log(M))$

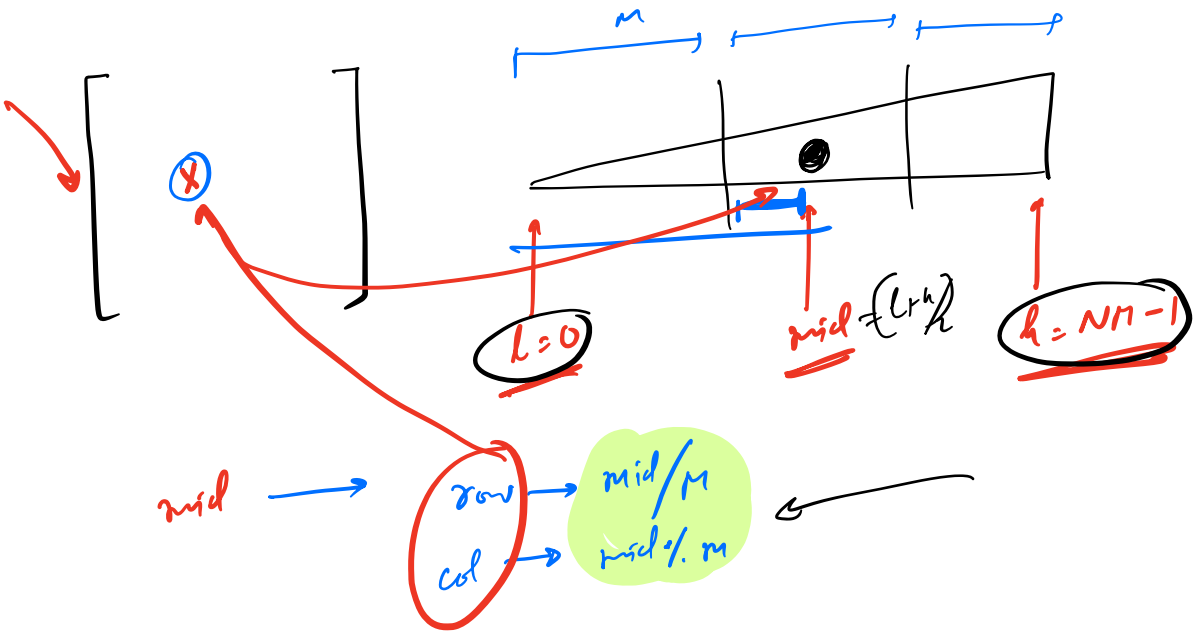
IV) find the first row s.t the last elmt of it $>= K$ using BS

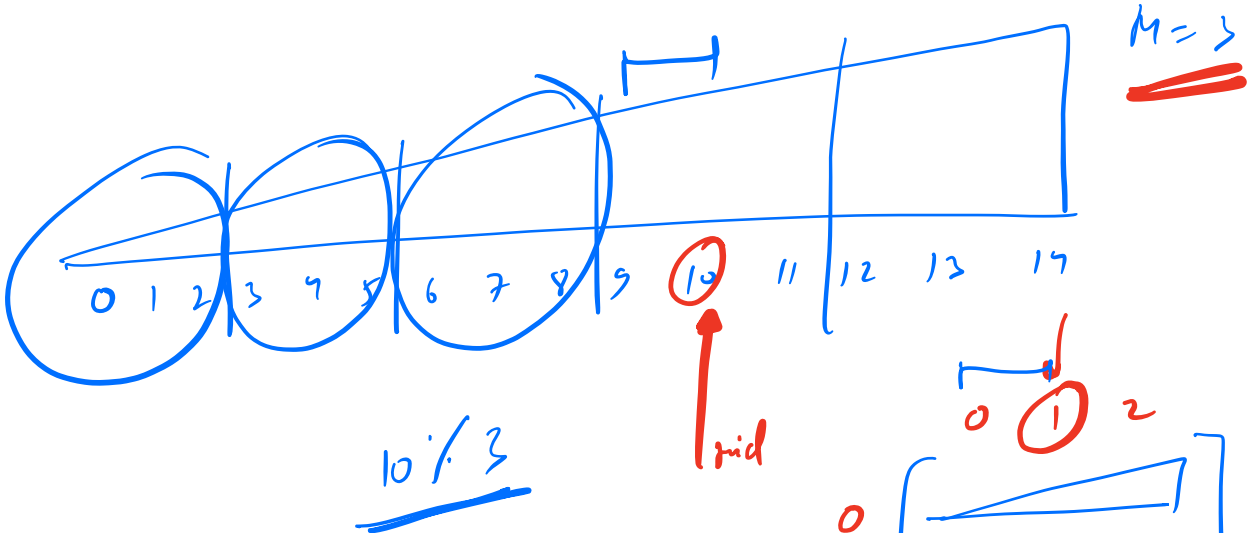
↳ BS in it

$O(\log(N) + \log(M))$

$\rightarrow \log(NM)$

m m





TC = O(log(NM))

SC = O(1)

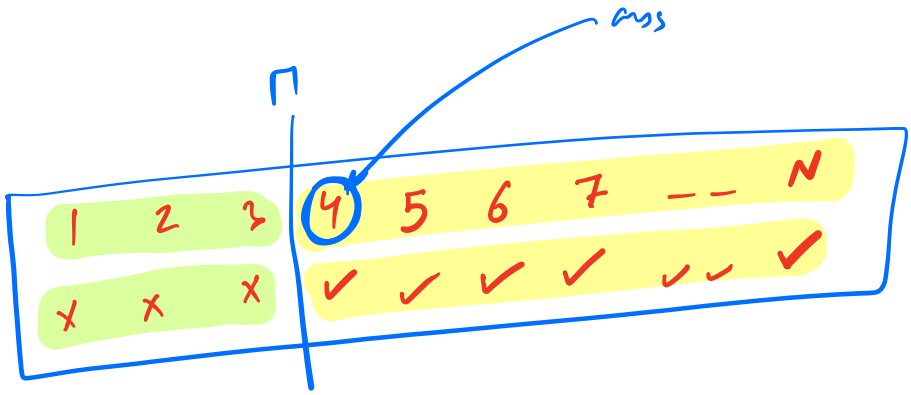
Q Given a string S. $S_i \in \{a, b, c\}$

Find the min. S.A. length

s.t. it contains all of $(a, b \& c)$

S: a a b a b b c c b b a

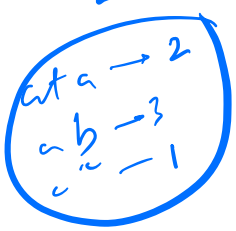
ANS: 4



6 ?

S: a a b a b b c c b b a

SLIDING WINDOW



```
//s   l = 3 , h = N , ans = -1        #it → O(log N)
while( l <= h) {                                    O(N)
    m = (l+h)/2;
    if( check(S, m) == true) {
        ans = m;
        h = m-1;
    }
    else {
        l = m+1;
    }
}
return ans;
```

SLIDING WINDOW

bool check if Any sub arry of size m contains all(a, b, c)

TC = O(N log N)

Q Given N. Find $\lfloor \sqrt{N} \rfloor$

$N \geq 1$

N: 27 → 5

30 → 5

35 → 5

36 → 6

49 → 7

48 → 6

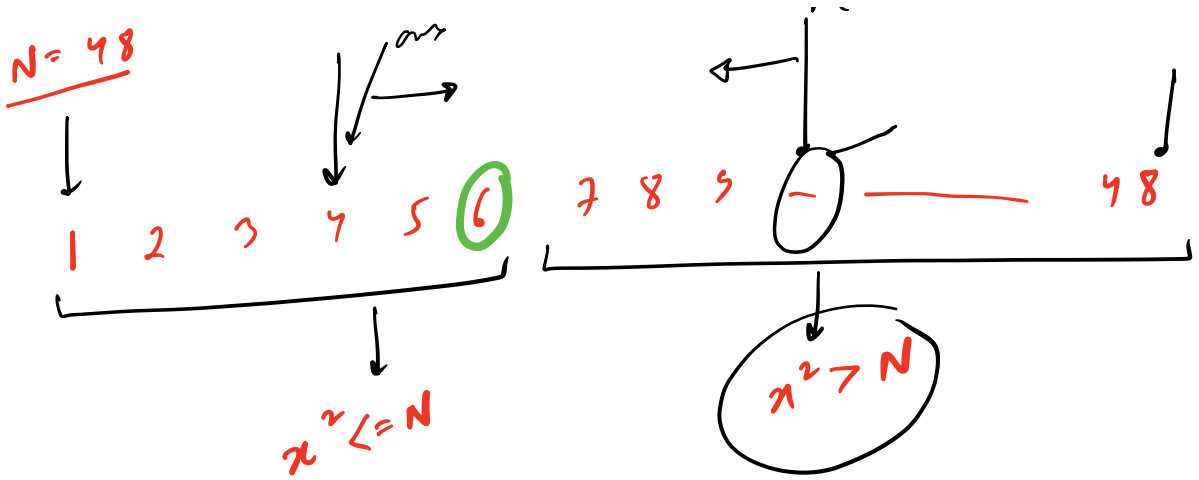
1. N: 48

ans

1 2 3 4 5 6 7 48

```
for (i:1 ——→ N) {
    if(i*i <= N) ans = i
    else break;
}
return ans;
```

TC = $O(\sqrt{N})$

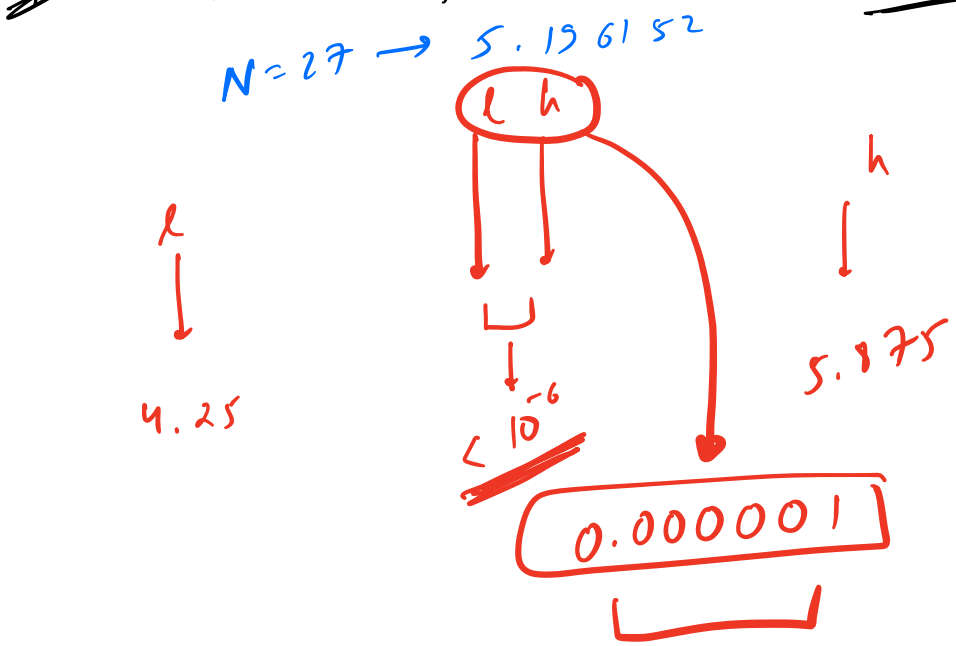


SS → N

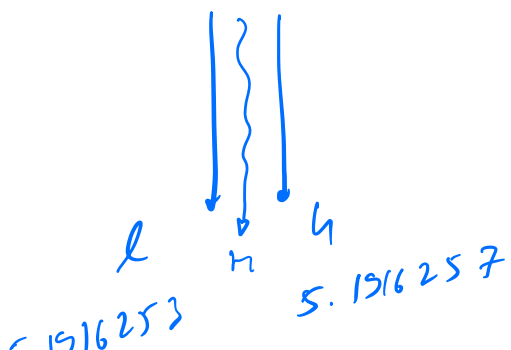
TC = O(log(N))

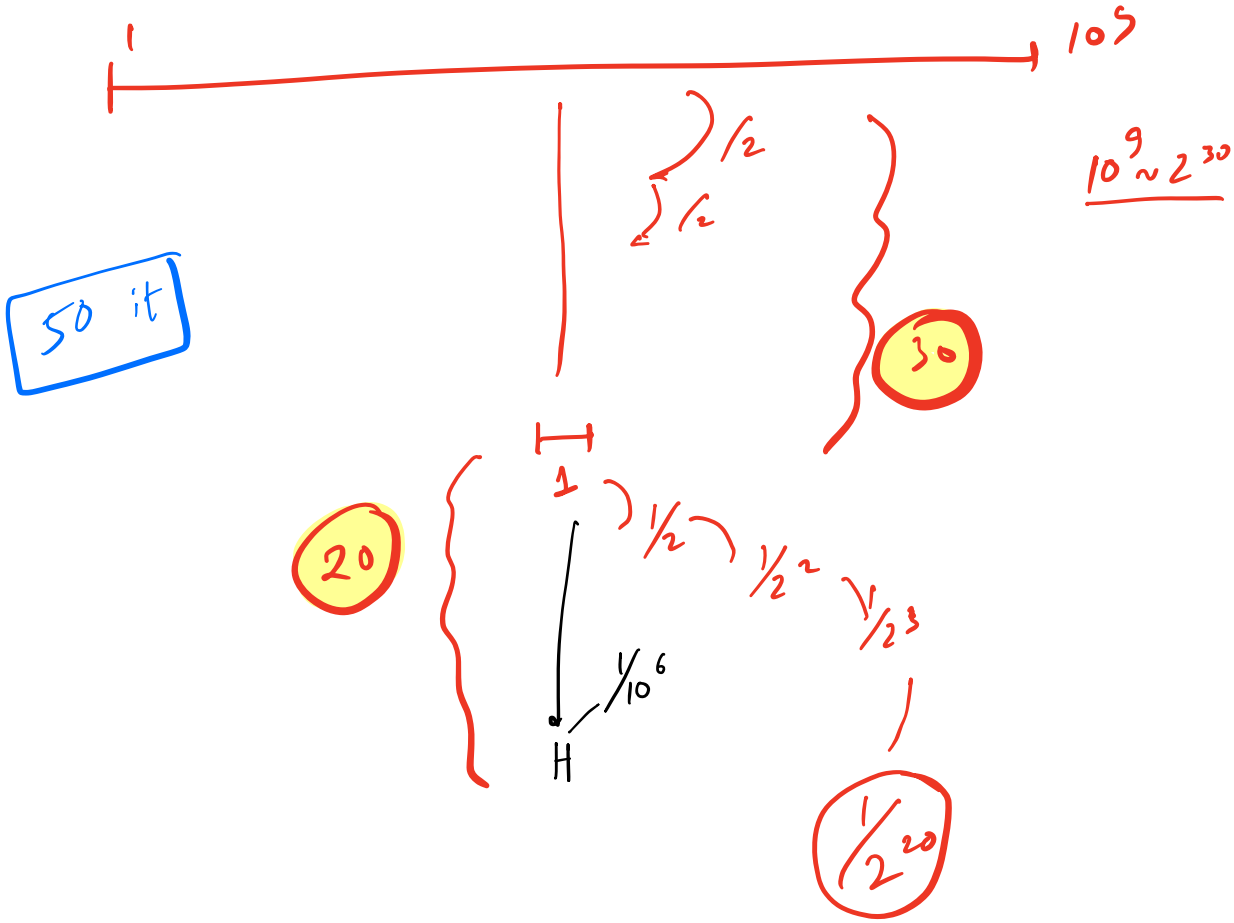
```
l = 1, h = N;
ans = -1;
while( l <= h) {
    m = (l+h)/2;
    if( m*m <= N) {
        ans = m;
        l = m+1;
    }
    else {
        h = m-1;
    }
}
return ans;
```

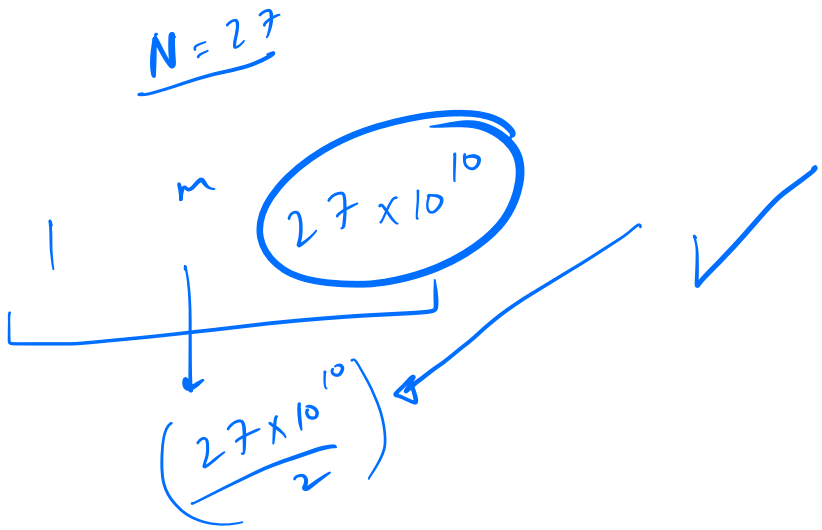
1 Given N. find $\sqrt{N}$. $1 <= N <= 10^9$




```
double l = 1, h = N;
while ( h - l > 0.000001 ) {
    double m = (l + h) / 2;
    if ( m * m <= N ) {
        l = m;
    }
    else {
        h = m;
    }
}
→ ret l, h, (l+h)/2
```




1
10^9
10^9 ~ 2^30
1/2
1/2
50 it
30
20
1
1/2
1/2^2
1/2^3
1/10^6
H
1/2^20


N = 27
1
m
27 x 10^10
(27 x 10^10 / 2)