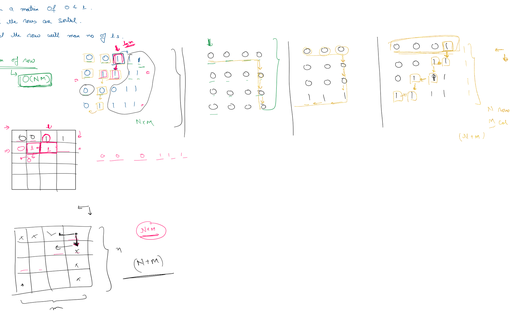
Problems on Arrays

Friday, 27 August 2021 9:03 PM

- Create
- Insert
- Delete

Q Given a matrix of 0 & 1.
All the rows are sorted.
Find the row with max no of 1s.

Sum of row — $O(N*M)$



$N \times M$

$(N+M)$

Q Given a matrix where every row & every col is sorted.
Search a no K in this matrix.

Average Expected Efficient

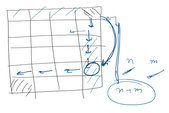
5	8	8.5	11
8	15	25	50
9	18	30	67

12

K = 2 → F
K = 37 → F

$(N+M)$

$N \times M$



$n+m$

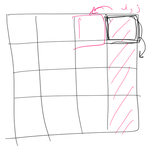
$N \times M$

→ Start from top right
$i = 0$;
$j = M-1$;

→ if $(a[i][j] == K)$ ret True;

→ else if $(a[i][j] > K)$
$j--$

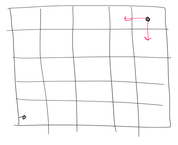
→ else
$i++$;



$i = 0$;
$j = M-1$;
While $(i < N \;\&\&\; j >= 0)$ { $j >= 0$ }
if $(a[i][j] == K)$
ret True;
else if $(a[i][j] > K)$
$j--$;
else // $a[i][j] < K$
$i++$;
}

$n <= (n-1)$

$O(M+N)$



N rows
M col

$(N+M)$

2 4 4 4
2 4 4 4
2 4 4 4

n m
10 32
$10 \times \log 32$
5
$\Rightarrow 50$

$32 + 10$
42

Sub array vs Sub seq
Sub string

1 2 4 3 10 7 ⇐ 1 3 10 7 _
4 10
4 3 10 _

c a b g x y c

2 7 1 10 9 11

1 _ 9 11 _

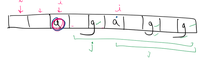
Q Given a string. Count no of sub seq "a g"

c a b g e g a x g l g s

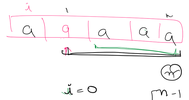
6

g a b a g

$O(N^2)$



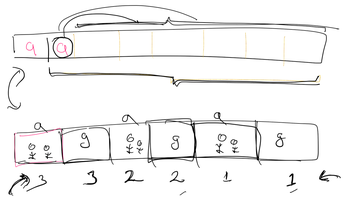
ans = 0;
for $(i = 0 \longrightarrow n)$ {
if $(a[i] == 'a')$ {
for $(j = i+1 \longrightarrow n)$ {
if $(a[j] == 'g')$
ans++;
}
}
}

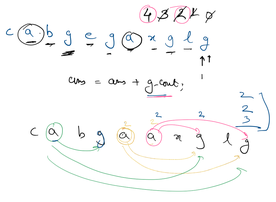


$i = 0$
$i = 1$
$i = 2$
⋮
$i = n-1$

$n-1$
$n-2$
$n-3$
⋮
0

$\simeq O(N^2)$





ans = ans + g_cnt;

ans = 0; g_cnt = 0;
for $(i = n-1;\ i >= 0;\ i--)$ {
if $(a[i] == 'g')$
g_cnt++;
else if $(a[i] == 'a')$ {
ans = ans + g_cnt;
}

a b c
2 × 2 × 2

2^n

a
b
c
ab
bc
ac
abc

Q Given an array. Find length of smallest sub-array which contain both max & min of the array.

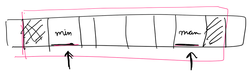
3, 2, 1, 6, 8, 4, 2, 1, 5, 4, 7, 8
0 1 2 3 4 5 6 7 8 9 10 11

max = 8
min = 1

⇒ 3

1, 2, 3, 1, 3, 4, 4

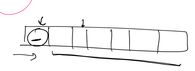
$O(N^2)$



⇒ Find the min-max pair closest to each other.

2, 5, 1, 4, 7, 8, 7, 8, 3, 2, 5, 7, 1

① Find Min
② Find Max



L-min
L-max

3, 2, 1, 6, 8, 4, 2, 1, 5, 4, 7, 8
0 1 2 3 4 5 6 7 8 9 10 11

J-Min : 1 O(N)
Max : 8 O(N)

i = 4
L-min = 2

R L
4 − 2 + 1

Latest-Min : ∞ 2
Latest-max : −∞
ans = ∞; 3

for (i = 0; i < n; i++) {
if (a[i] == min) {
ans = min(ans, 3, i − L-max + 1);
L-min = i;
}
else if (a[i] == max) {
ans = min(ans, i − L-min + 1);
L-max = i;
}
}

$O(N) + O(N)$
$\simeq O(N)$

min : 1
max : 10

10, 10, 2, 4, 3, 1, 1, 4, 10, 1, 1
10

L-min
L-max
ans = ∞ ~~5~~ ~~3~~ 2