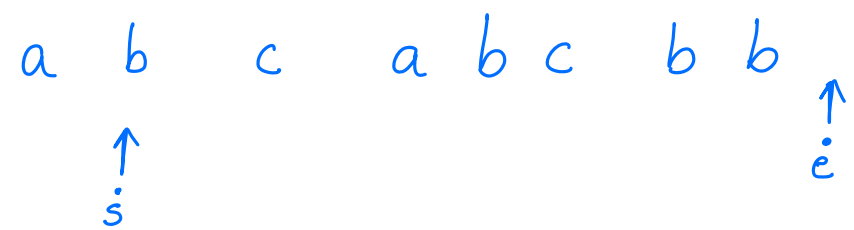
1. Longest substring without repetition

a b c a b c b b

A a a A

{ Acquire
and
release method
}



s e

s to e-1

{ bc a }

acquire [e++ → valid ✓

longest valid sub.

release [s++

[while (e < N) {
while (____) {
acquiring →

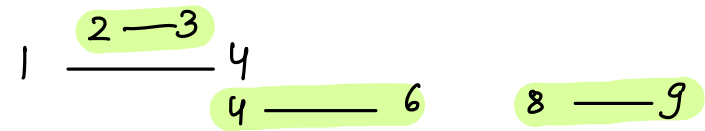
}

while (____) {
releasing

} }

Q2 Disjoint sets

A = { (1, 4)
(2, 3) ✓
(4, 6) ✓
(8, 9) ✓



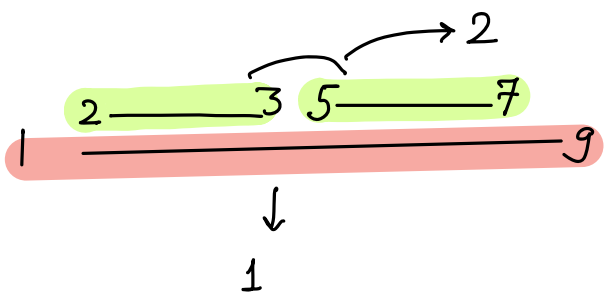
s_1 e_1 s_2 e_2

}

$e_1 < s_2$ [nothing is common]

Ex 2

{ [1, 9
2, 3
5, 7] }



Sorting

[(1, 3) (4, 5) (8, 9) (1, 9)]

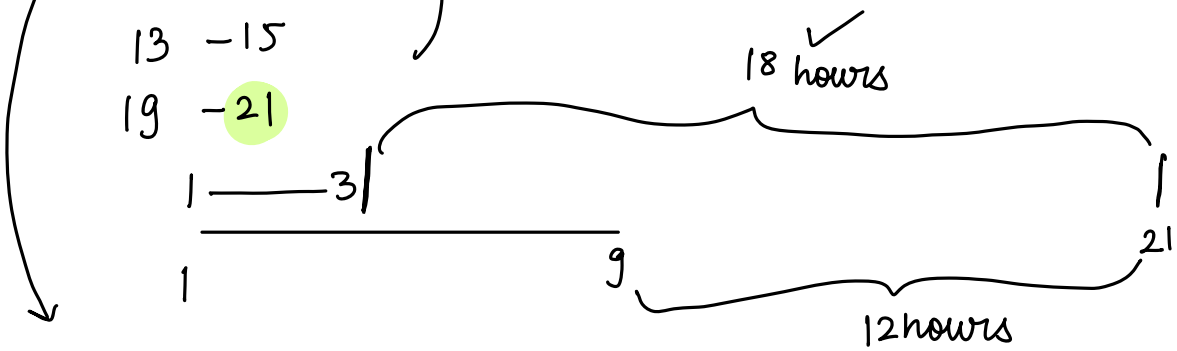
Sorting on basis of end point

max = 23:59

1—3

1 3
4 5
1 - 9
7 - 8
11 - 12
14 - 17
13 - 15
19 - 21

more meetings

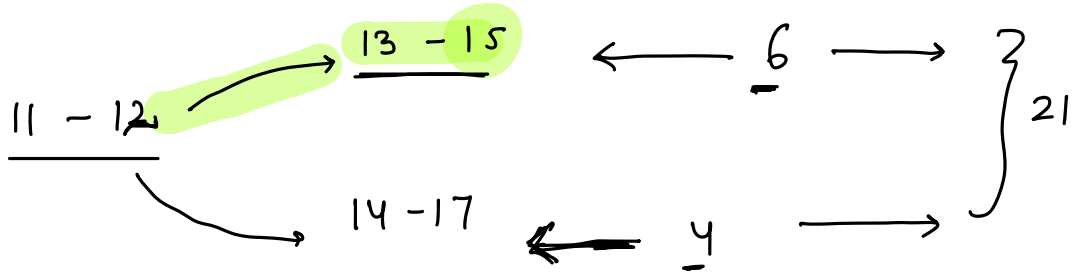


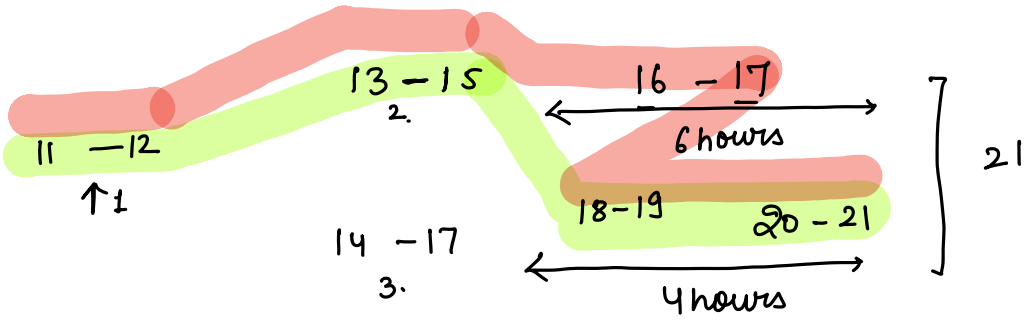
1-3 4 5 7 - 8 1 - 9 11 - 12 13-15 14 -17 19-21

1-3 4-5 7-8 11 -12 13 - 15 19-21

e1 < s2

(s1, e1) (s2, e2)





• 11 12 | 13 15 | 14 17 | 16 19 | 18 19 | 20 21

11 -12 | 13 -15 | X | 16 -19 | X | 20 21

1 Sort on basis ending point

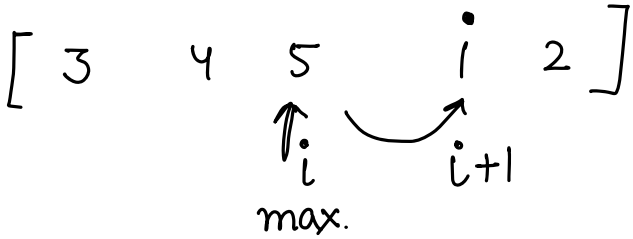
2. Keep taking intervals which are ending first & non-clashing

3. find longest length.

Q3 Sorted and rotated array

↓

Search

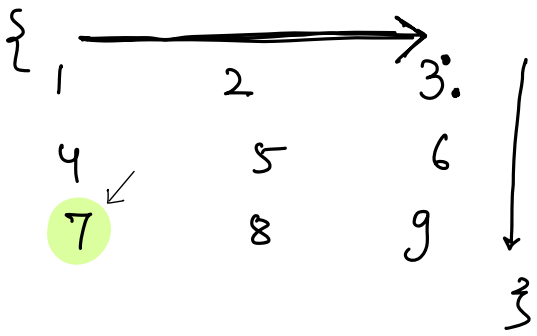


Q4. (1 buy – 1 Sell)

1 gold	30	50	40	20	50	10	60 →
	M	T	W	T	F	S	S
	60	60	60	60	60	60	
Profit	30	10	20	40	10	50	

{ buying on A[i]
Sell → max (A[i+1] till A[N-1])] - Diff { Profit }

Q 5



(7)

ele < 3
←
change col j--

ele > 3
i++

1 2 2
3 3 4
3 4 5

3
↳ i j → min

i x 1001 + j ✓ ← j-- ← j >= 0

i < N