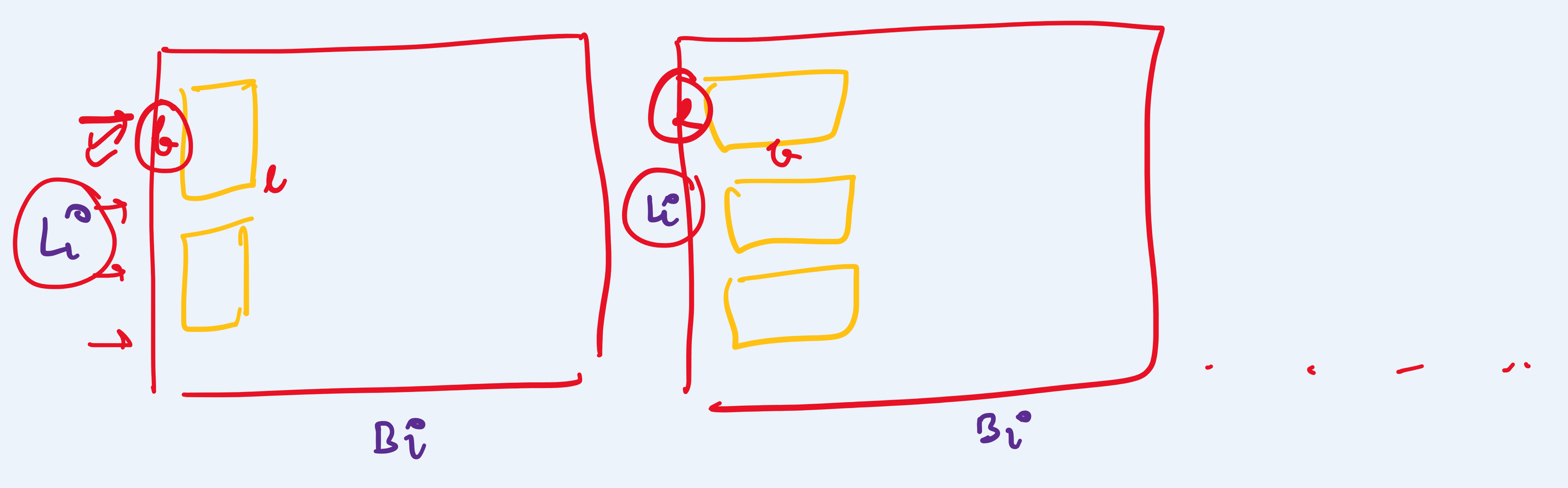
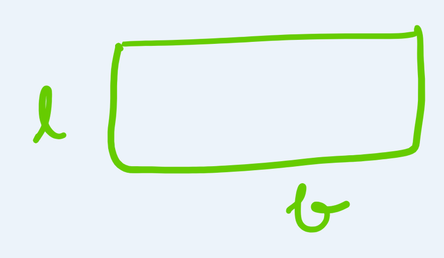
Any doubts ??

Q Codechef — (RAMDEV)

N halls





$$\sum_{i=0}^{n-1} \max\left(\frac{Li}{l} \times \frac{Bi}{b}, \frac{Li}{b} \times \frac{Bi}{l}\right)$$

mal → l, b

result = 0

```
for ( i= 0; i< n; i++) {
        int  Li =  scn. nextInt();
        int  Bi =  scn. nextInt();

        result += max((Li/l)*(Bi/b), (Li/b)*(Bi/l));
```

Q Meeting rooms II

intervals [i] = [start i, end i]

minimum no. of conference rooms required.

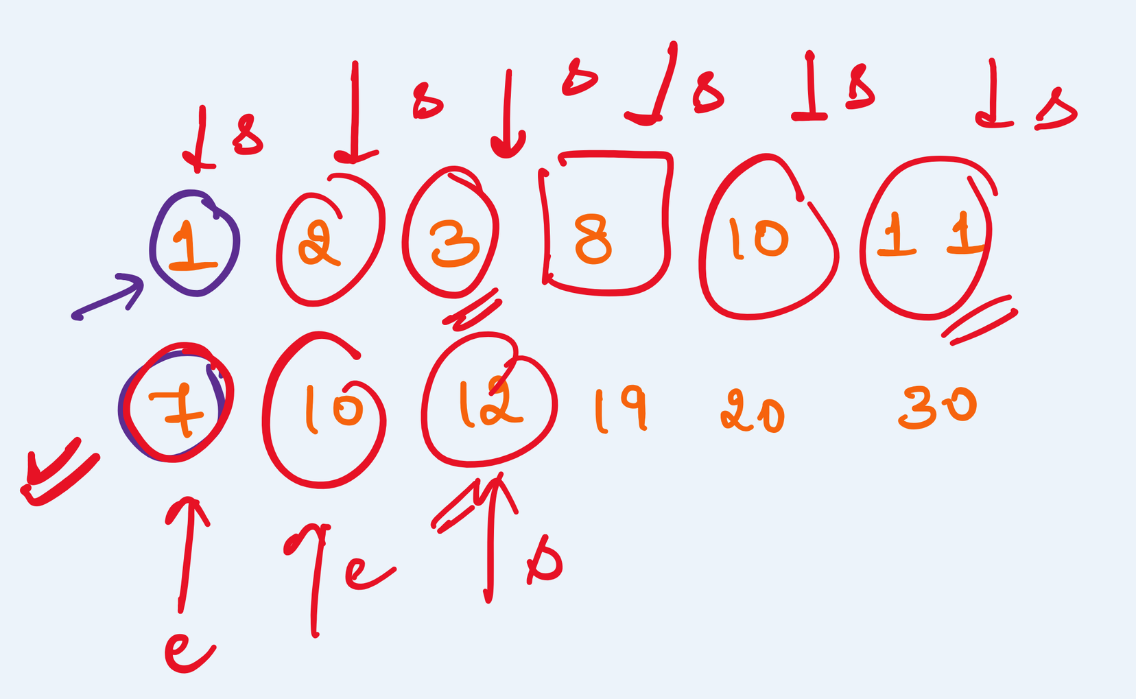
(1, 10), (2, 7), (3, 19), (8, 12), (10, 20), (11, 30)

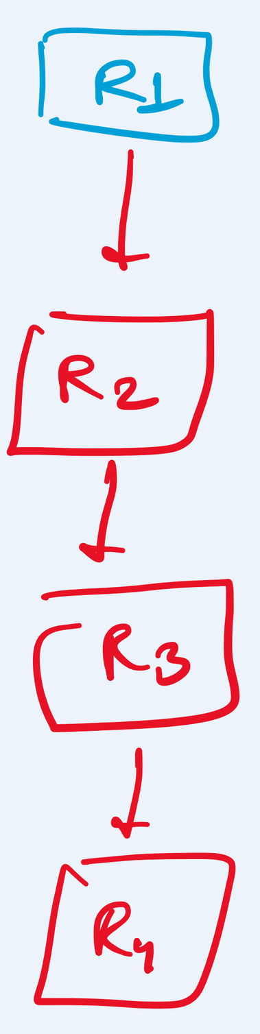
start

(1, 2, 3, 8, 10, 11)

end

(7, 10, 12, 19, 20, 30)

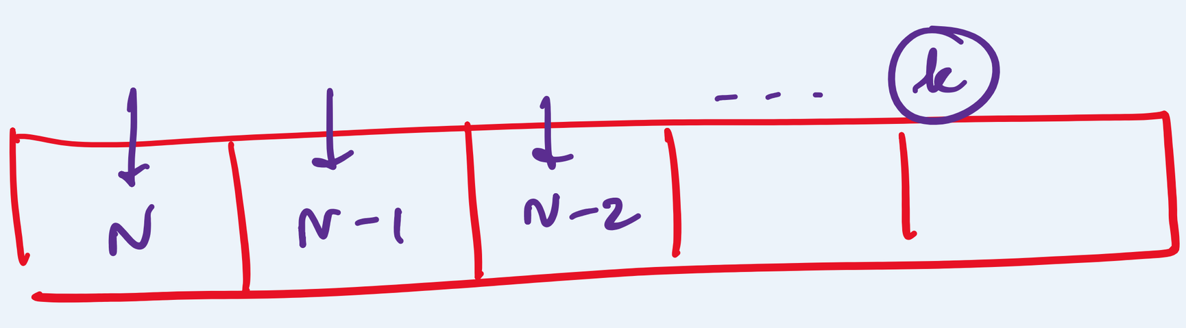




Q Given an array (unsorted) containing the first N natural no. of length N. Given a no k, find the Longest permutation of the array you can build by applying k- operations. (at max) on it. In an operation, you can choose any 2 elements & swap them.

Eg: {1, 2, 3, 4} , k = 1

ans → 4, 2, 3, 1



1 — N

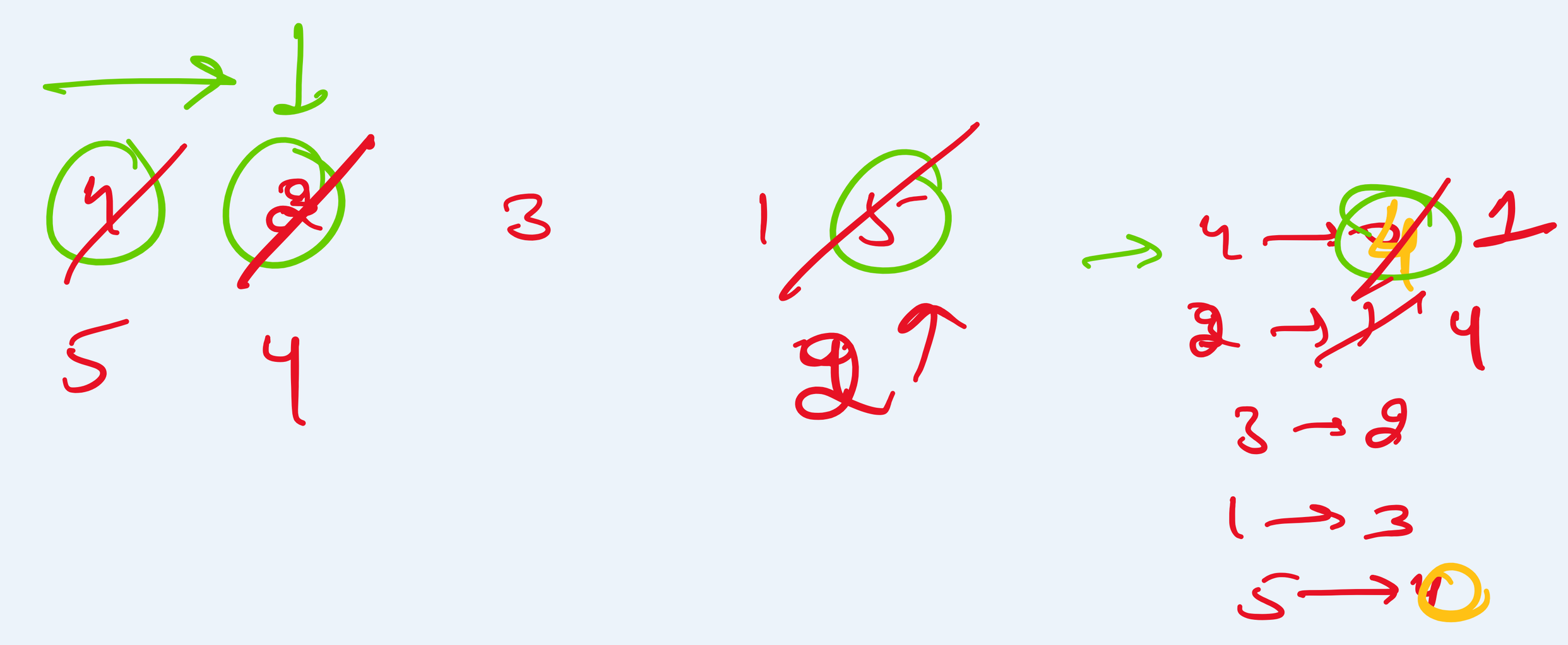
first N natural no.

We already know the biggest & smallest no.

N N-1 N-2 ..

map →

(key ele) value index




```
for ( i=0; i<n; i++) {
        input (arr[i]);
        map [arr [i]] =  i ;

}
input (k)

el = n
while (k--) {
      int   curr = el;
      int   curr_idx  =  map [curr];
      int   correct_index =  n- curr;

      int   el_to_be_swapped =  arr [correct_idx];
      swap (  arr [curr_idx],  arr [correct_idx]);

      map [curr] =  correct_idx
      map [el_to_be_swapped] =  curr_idx;
      el--;
}
```

Q Maximise sum after k negations

[2, -3, -1, 5, -4] — k = 2

output → 13

-4 → 4

[2, -3, -1, 5, 4]

-3 → 3

[2, 3, -1, 5, 4]