PAINTER'S PARTITION

☆ In this problem, we are given an array of integers where each integer represents units of time to be painted along with K no. of painters

We are supposed to return what is the least amount of time it would take all the painters to finish. i.e, we split array in K halves & find maximum sum which is least overall

Each painter must have something to do.

Solution for this is exactly similar to book allocation question hence the answer is same.

Pseudocode :

```
paintSplitting (arr, N, m, s) {
    k = 1, cu = 0
    for (i = 1 → N) {
        if (arr[i] + cu > m) {
            k++
            cu = arr[i]
        } else  cu += arr[i]
    }
    return k
}
```

```
paintersPartition(arr, N, s) {
    max, sum = 0, 0
    for(i = 0 → N) {
        if (arr[i] > max) {
            max = arr[i]
        }

        sum + = arr[i]
    }

    ans = -1
    while (max <= sum) {
        mid = (max + sum)/2
        ret = paintSplitting(arr, N, mid, s)
        if (ret < s) {
            sum = mid - 1
        } else {
            max = mid + 1
        }
    }

    return max
}
```