

LeetCode #2600

→ Day-10

('K') Items with the Maximum Sum

Example

numOnes = 3

numZeros = 2

K = 2

numNegOnes = 0

then Bag of Items $\Rightarrow \{1, 1, 1, 0, 0\}$

we have to take two items (K)
from this bag which would make the
maximum sum possible. which in this

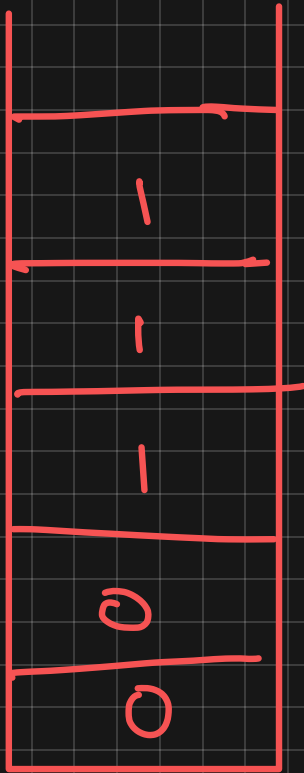
Case is (2) $[1+1]$

Approach (slow)

- ① We take the greedy approach and use a priority queue from C++ STL.
- ② In a priority queue, the list will always be sorted either in ascending or in descending order. By default in C++, it's stored in a non-decreasing order making the top element the biggest/greatest element, and then by being greedy, we can take ' k ' elements from the list and sum them up.

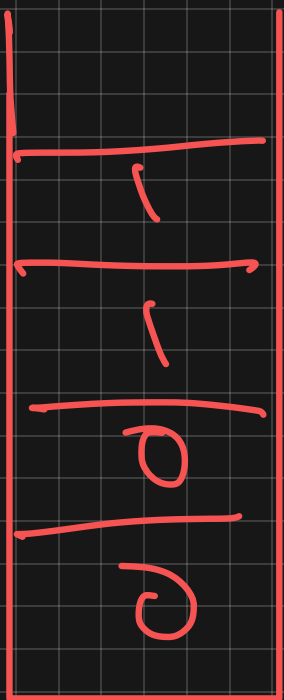
Illustrations

`int sum = 0; k = 2`



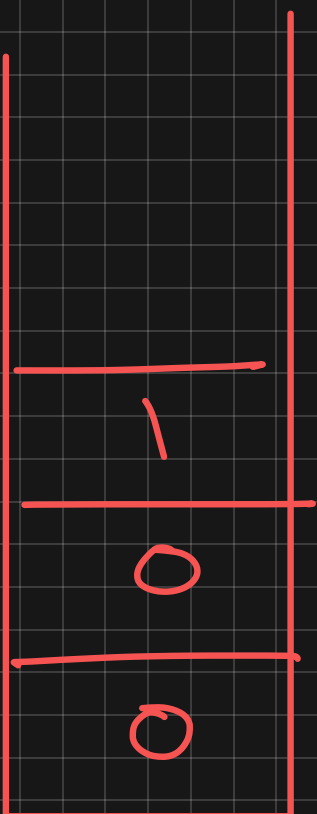
→ take the Top element
add to sum.
then pop it

1 time



→ Take the top element, add it
to sum. then pop it.

2 time



Total sum = 2

Approach (FAST)

- ① Using just if and else statements and for every total number of 1's, 0's and -1's provided, add them to sum.