Q. 3

Using a sparse representation, each column of the matrix is a TreeSet<Integer> instance

★ Cost of the substraction operation:

To substract two columns, you substract the two TreeSet by traversing the TreeSet to substract, and, for each of its Integer, either the Integer is in the treeSet, and therefore you remove it, either it's not, and you add it.

Looking, inserting or removing in a TreeSet is a $O(\log n)$ cost operation.

But if we assume that both columns are sparse, then it is in $O(1)$ for each insertion and there are $O(1)$ insertion, so in the end: $O(1)$ is the cost of substraction

★ How many substractions?

There can be at most m substractions to do: $O(m)$

★ How many lookups? (cost of a lookup)

Instead of looping over the columns, we use an HashMap from the lowest non null indices to the TreeSet.

Every lookup therefore costs $O(1)$

★ How many columns on which to perform these steps?

There are m of them: $O(m)$

All in all the complexity is: $O(m^2)$