▣ Description	△ Solution	🗩 Discuss (999+)	🕘 Submissions

Example :

```
Let the string be  : abbccdddeeffffggg
So count of a=1, b=2, c=2, d=3, e=2, f=4, g=3. So freq = {1,2,2,3,2,4,3,0,......,0}
So after sorting the elements of the freq array are = {0,......, 0, 1,2,2,2,3,3,4}

So at each step in the loop : [ Element of consideration is included in ' ', ans next e

[0, .... ,0,1,2,2,2,3,'3',(4)] -> '3' != 0, '3' < (4), So continue.
[0, .... ,0,1,2,2,2,'3',(3),4] -> '3' != 0, '3' == (3), So prev = '3', freq[i] = max(0,
[0, .... ,0,1,2,2,'2',(2),3,4] -> '2' != 0, '2' == (2), So prev = '2', freq[i] = max(0,
[0, .... ,0,1,2,'2',(1),2,3,4] -> '2' != 0, '2' > (1), So prev = '2', freq[i] = max(0,
[0, .... ,0,1,'2',(0),1,2,3,4] -> '2' != 0, '2' > (0), So prev = '2', freq[i] = max(0,
[0, .... ,0,'1',(0),0,1,2,3,4] -> '1' != 0, '1' > (0), So prev = '1', freq[i] = max(0,
[0, .... ,'0',(0),0,0,1,2,3,4] -> '0' == 0, break;

See all frequencies become distinct so return del i.e. 7
```

Code with Full Explanation :

```cpp
class Solution {
public:
    int minDeletions(string s) {
        //Array to store the count of each character.
        vector<int> freq (26, 0);

        //Calculatimg frequency of all characters.
        for (char c : s){
            freq[c - 'a']++;
        }

        //sorting the frequencies. So the greatest frequencies are in right side.
        sort(freq.begin(), freq.end());

        int del = 0; //to store the deletions.

        //Checking if 2 frequencies are same, if same then decrease the frequency so th
        for (int i = 24; i >= 0; i--) {

            if(freq[i] == 0) break; // if frequency is 0 that means no more character i

            if(freq[i] >= freq[i+1]){

                int prev = freq[i]; //To store the frequency before deletion.
                freq[i] = max(0, freq[i+1] -1); //New frequency should be 1 less than t
```