Int with Maximum Freq

arr = [5, 2, 5, 3, 5, 3, 0, 5, 2]

i

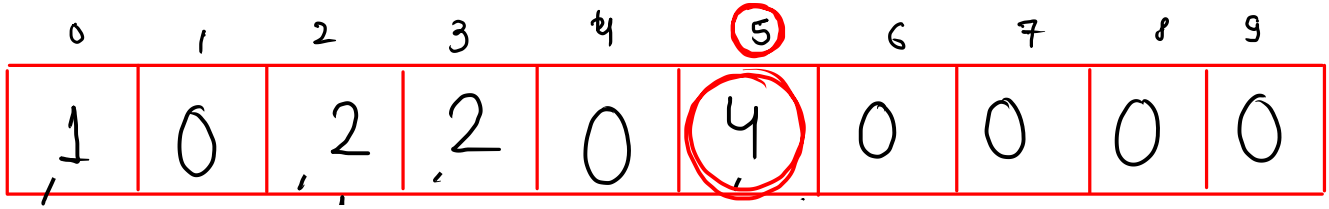
freq (int) =

0	1	2	3	4	5	6	7	8	9
1	0	2	2	0	4	0	0	0	0

```
int idx = arr[i];
freq[idx]++;
```

second loop

freq = (int)



i

maxValue = ~~-∞~~ ~~1~~ ~~2~~ 4

index = ~~-1~~ ~~0~~ ~~2~~ 5

Code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }
    System.out.println(findMaxInt(arr, n));
}
public static int findMaxInt(int[] arr, int n) {
    int[] freq = new int[10];
    for (int i = 0; i < n; i++) {
        int idx = arr[i];
        freq[idx]++;
    }

    int maxValue = Integer.MIN_VALUE;
    int index = -1;
    for (int i = 0; i < 10; i++) {
        if ( freq[i] > maxValue ) {
            maxValue = freq[i];
            index = i;
        }
    }
    return index;
}
```

O(n)

O(1)

T.C = O(n)

S.C = O(1)

Maximum Freq Character

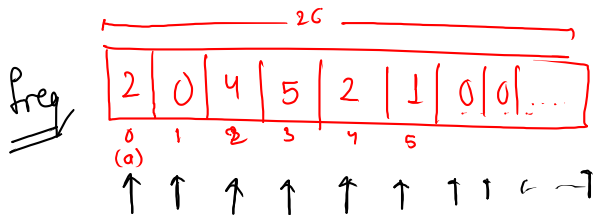
```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    System.out.println(findMaxChar(str));
}
public static char findMaxChar(String str) {
    int[] freq = new int[26];
    for (int i = 0; i < str.length(); i++) {
        char ch = str.charAt(i);
        int idx = ch - 'a';
        freq[idx]++;
    }

    int maxValue = Integer.MIN_VALUE;
    char ch = '+';
    for (int i = 0; i < 26; i++) {
        if ( freq[i] > maxValue ) {
            maxValue = freq[i];
            ch = (char)(i + 'a');
        }
    }
    return ch;
}
```

3 + 'a'
3 + 97
(char) 100 → 'd'

T.C = O(n)

S.C = O(1)



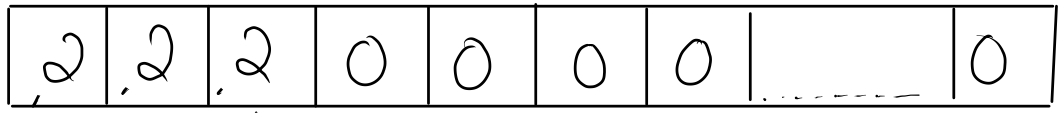
maxValue = ~~-∞~~ 2 4 5
ch = ~~'+'~~ ~~'c'~~ ~~'c'~~ 'd'

Good String Checker

String str = "abcbac";

0 1 2 3 4 5

freq = (int)

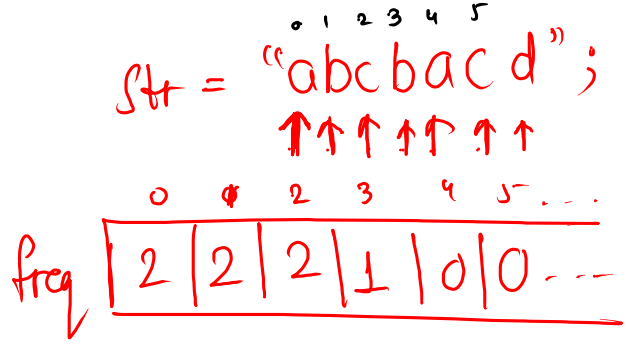


val = freq[idx];

Code

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    System.out.println(goodStringChecker(str));
}
public static boolean goodStringChecker(String str) {
    int[] freq = new int[26];
    for (int i = 0; i < str.length(); i++) {
        char ch = str.charAt(i);
        int idx = ch - 'a';
        freq[idx]++;
    }

    char ch = str.charAt(0);
    int idx = ch - 'a';
    int f = freq[idx];
    for (int i = 0; i < str.length(); i++) {
        char ch1 = str.charAt(i);
        int idx1 = ch1 - 'a';
        if ( freq[idx1] != f ) {
            return false;
        }
    }
    return true;
}
```

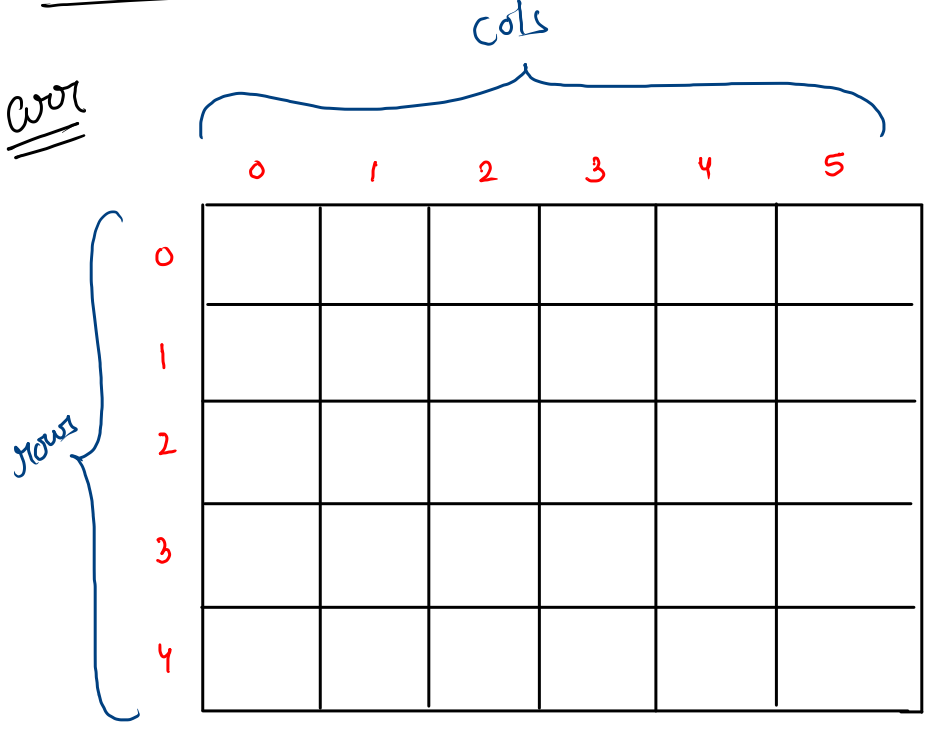


T.C = O(n)

S.C = O(1)

⇒ 2D array

rows → horizontal
cols → vertical



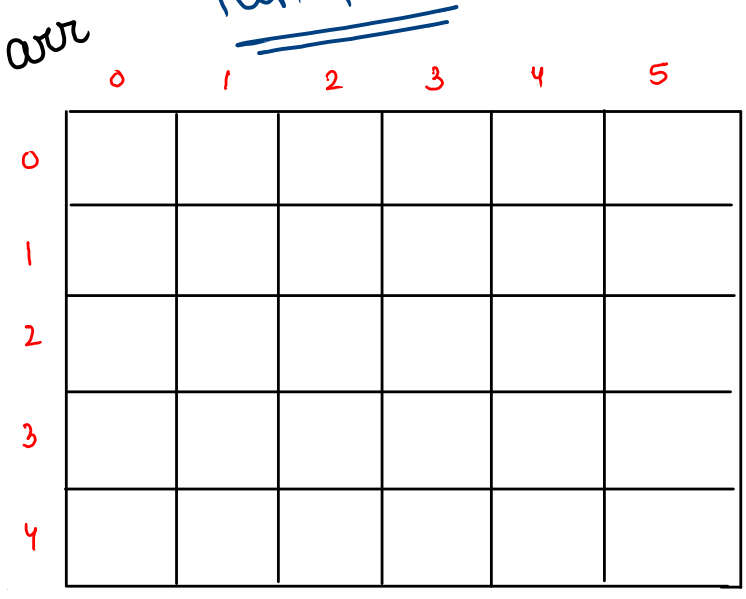
rows = 5
cols = 6

size = 5 × 6

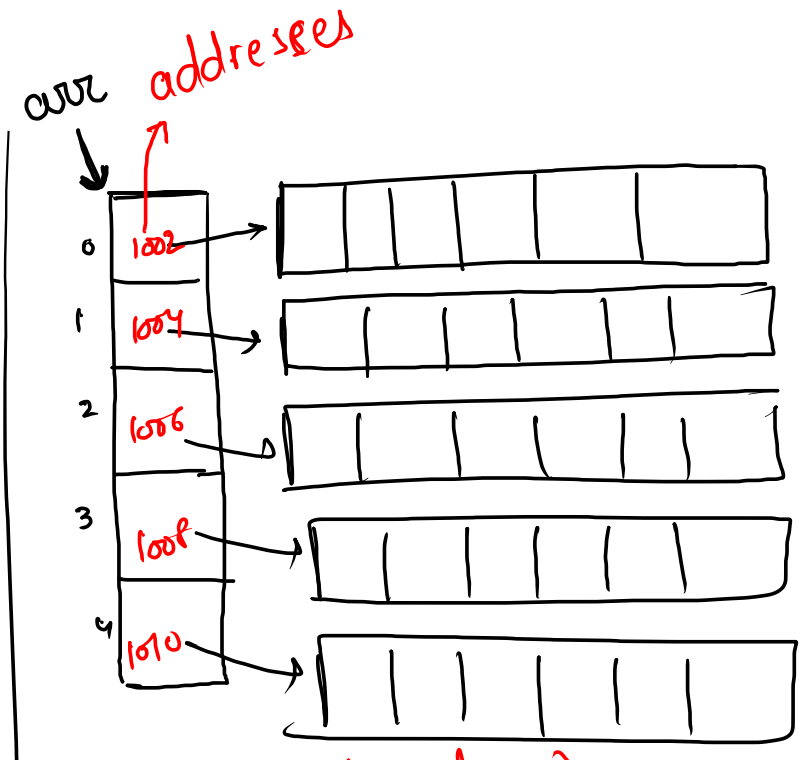
Note:-

rows = arr.length; (no. of rows) // 5
cols = arr[0].length; (no. of cols) // 6

fun fact



(we see)



(actually)

```
rows = arr.length;    (no. of rows)
cols = arr[0].length; (no. of cols)
```

arr

	0	1	2	3	4	5
0	5	2	-3	-100	0	7
1	0	2	3	1	8	9
2	7	8	10	12	15	19
3	25	-25	~~-25~~ 20	18	21	22
4	0	1	2	3	4	5

declare 2D array

```
int[][] arr = new int[rows][cols.];
```

```
rows = arr.length; (no. of rows)
cols = arr[0].length; (no. of cols)
```

how to access element

index = 2, 3

```
int a = arr[2][3];
```

upgradation

```
arr[3][2] = 20;
```

row index — col. index