

Modify The Matrix

Size = 8 x 16

arr

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

~~approach 1~~

by creating a copy of original
2d array

↳ traversing in 1 array and
updating in another

T.C = $O(m * n)$, S.C = $O(m * n)$

Approach 2

cols

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

rows

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

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

i
j
 $(1, 2)$
 $(3, 7)$
 $(7, 4)$

$$T.C = O(m \times n), \quad S.C = O(m + n)$$

pseudo
code

i/p 2D array ($m \times n$)

- 1) create 1d array of size m // rows
 - 2) create 1d array of size n // cols
 - 3) traverse in 2d array
 - 3.1) check if $\text{arr}[i][j] == 1$
update $\text{rows}[i] = 1$
 $\text{cols}[j] = 1$
 - 4) traverse in 2d array
 - 4.1) check if $\text{row}[i] == 1$ or $\text{cols}[j] == 1$
update $\text{arr}[i][j] = 1$
- return 2d array

Code

S.C = O(m+n) , T.C = O(m*n)

```
public static int[][] modifyMatrix(int[][] arr, int m, int n) {  
    int[] rows = new int[m];  
    int[] cols = new int[n];  
    for (int i = 0; i < m; i++) {  
        for (int j = 0; j < n; j++) {  
            if (arr[i][j] == 1) {  
                rows[i] = 1;  
                cols[j] = 1;  
            }  
        }  
    }  
    for (int i = 0; i < m; i++) {  
        for (int j = 0; j < n; j++) {  
            if (rows[i] == 1 || cols[j] == 1) {  
                arr[i][j] = 1;  
            }  
        }  
    }  
    return arr;  
}
```

⇒ Strings

str = "Abcdefg";
 0 1 2 3 4 5 6

inbuilt

- 1) str.length(); // 7
- 2) str.charAt(index); // 'd'
- 3) str.toUpperCase(); // "ABCDEFG"
- 4) str.toLowerCase(); // "abcdefg"

str = "A**c**d**e**f**g**" ; substring :- sub part of a string
 0 1 2 3 4 5 6

Syntax :- str.substring (start index , end index + 1) ;

- 1) str.substring (1 , 5) ; // bcde
- 1) str.substring (0 , 5) ; // Abcde
- 1) str.substring (3 , 5) ; // de
- 1) str.substring (3 , 4) ; // d
- 1) str.substring (4 , 6) ; // ef
- 1) str.substring (4 , 7) ; // efg
- 1) str.substring (4 , 8) ; // error

Syntax 2

$\text{str} = \text{"Abcdefg"}$

$\text{str.substring(start index)}$

$\text{str.substring(4)} // \text{efg}$

Print Characters

```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    String str = scn.nextLine();  
  
    for (int i = 0; i < str.length(); i++) {  
        System.out.println( str.charAt(i) );  
    }  
}
```

Is Equal?

observation

- ↳ length of both strings should be same
- ↳ each char at same index should also be same

Code

T.C = O(str1.length())

```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    String str1 = scn.nextLine();  
    String str2 = scn.nextLine();  
  
    System.out.println(checkEqual(str1, str2));  
}  
public static boolean checkEqual(String str1, String str2) {  
    if ( str1.length() != str2.length() ) {  
        return false;  
    } else {  
        for (int i = 0; i < str1.length(); i++) {  
            if ( str1.charAt(i) != str2.charAt(i) ) {  
                return false;  
            }  
        }  
        return true;  
    }  
}
```

Note:-

strings can't be compared

with ==

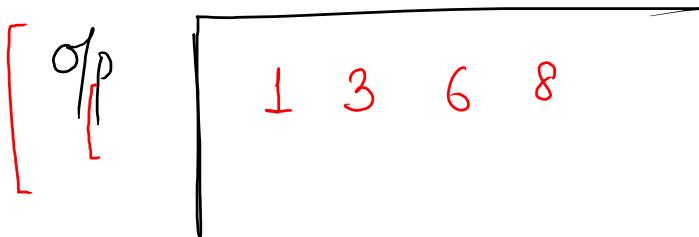
instead use str1.equals(str2);

code

```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    String str1 = scn.nextLine();  
    String str2 = scn.nextLine();  
  
    boolean ans = str1.equals(str2);  
    System.out.println(ans);  
}
```

Print Indices of Vowels

str = "KunalSuri";
0 1 2 3 4 5 6 7 8



```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    String str = scn.nextLine();  
  
    for (int i = 0; i < str.length(); i++) {  
        char ch = str.charAt(i);  
        if (isVowel(ch)) {  
            System.out.print(i + " ");  
        }  
    }  
}  
public static boolean isVowel(char ch) {  
    return ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' || ch == 'A' || ch == 'E' ||  
    ch == 'I' || ch == 'O' || ch == 'U';  
}
```