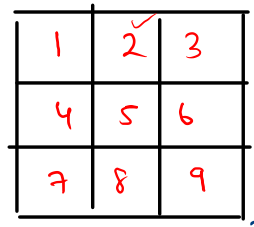
Add Two Matrices

Sample Input 0

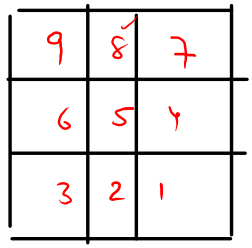
```
3 3
1 2 3
4 5 6
7 8 9
3 3
9 8 7
6 5 4
3 2 1
```



3x3
m x n

(A)

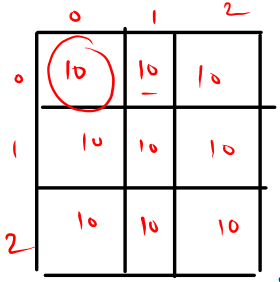
+



3x3
p x q

(B)

=



3 x 3
s x t

(C)

cond^n ?

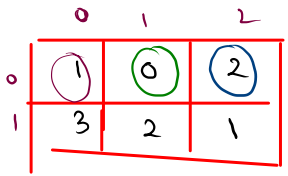
m == p
n == q

} → dimensions should be same

C[i][j] = A[i][j] + B[i][j]

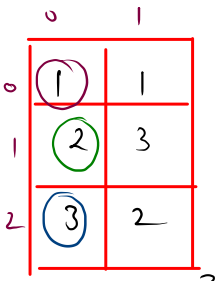
```java
import java.io.*;
import java.util.*;

public class Solution {
    public static void printArr(int [][] A){
        for(int [] d : A){
            for(int e : d){
                System.out.print(e + " ");
            }
            System.out.println();
        }
    }

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int m = scn.nextInt();
        int n = scn.nextInt();

        int [][] A = new int[m][n];

        for(int i = 0; i < m; i++){
            for(int j = 0; j < n; j++){
                A[i][j]  = scn.nextInt();
            }
        }

        int p = scn.nextInt();
        int q = scn.nextInt();

        int [][] B = new int[p][q];
        for(int i = 0; i < p; i++){
            for(int j = 0; j < q; j++){
                B[i][j]  = scn.nextInt();
            }
        }

        //logic

        if(m == p && q == n){
            //add
            int [][] C = new int[m][n];

            for(int i = 0; i < m; i++){
                for(int j = 0; j < n; j++){
                    C[i][j] = A[i][j] + B[i][j];
                }
            }

            printArr(C);


        }
        else{
            System.out.println("-1");
        }
    }
}
```

Multiplication of Matrix.



2×3

(A) $m \times n$



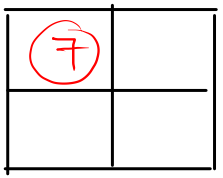
3×2

(B) $p \times q$

$\Rightarrow$ C

$m \times q$

$n == p$ ✓ → true ⇒ multiplication



2×2

$C[0][0] = A[0][0] * B[0][0]$
$+ A[0][1] * B[1][0]$
$+ A[0][2] * B[2][0]$

$= 1 * 1$
$+ 0 * 2$
$+ 2 * 3$

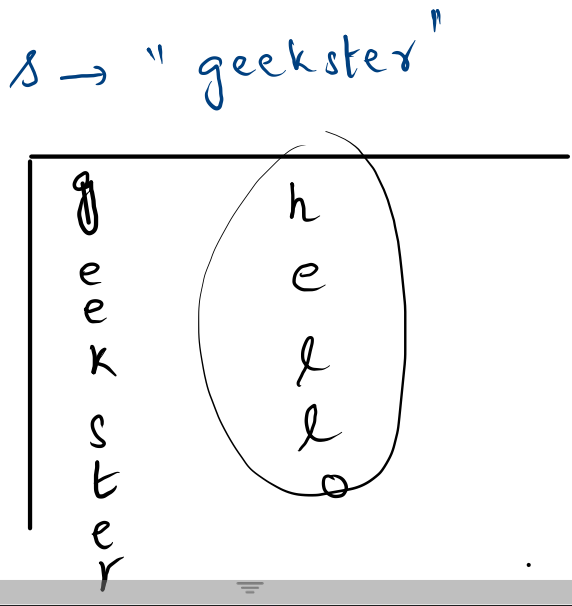
$= 7$

String → Non-primitive Data Type.

↳ sequence of character.

↳ * Is string a mutable or **im-mutable** data structure in Java.

Print Characters




```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);

    String s = scn.next();

    for(int i = 0; i < s.length(); i++){
        System.out.println(s.charAt(i));
    }

}
```

Is Equal?

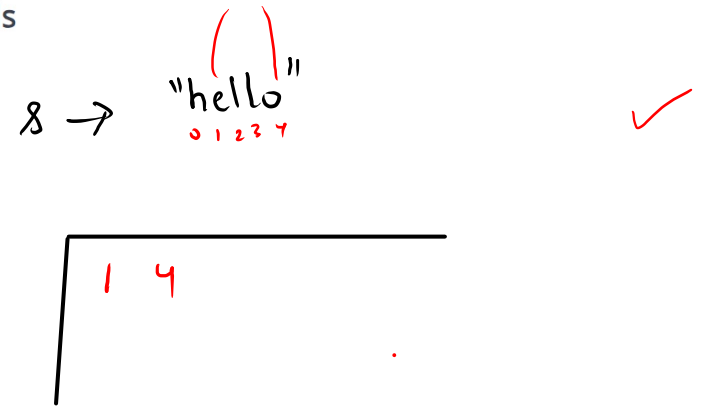
s → "hello"

t → "hello"

s.equals(t)

```java
import java.io.*;
import java.util.*;

public class Solution {
    public static boolean isEqual(String s, String t){
        if(s.length() != t.length()){
            return false;
        }

        for(int i = 0; i < s.length(); i++){
            if(s.charAt(i) != t.charAt(i)){
                return false;
            }
        }

        return true;
    }

    public static void main(String[] args) {
         Scanner scn = new Scanner(System.in);

        String s = scn.next();
        String t = scn.next();

        // System.out.println(s.equals(t));

        boolean ans = isEqual(s, t);
        System.out.println(ans);
    }
}
```

Print Indices of Vowels




```java
import java.util.*;

public class Solution {
    public static boolean isVowel(char ch){
        if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u'){
            return true;
        }
        return false;
    }

    public static void main(String[] args) {
         Scanner scn = new Scanner(System.in);

         String s = scn.next();

        for(int i = 0; i < s.length(); i++){
            char ch = s.charAt(i);
            if(isVowel(ch)){
                System.out.print(i + " ");
            }
        }
    }
}
```

125. Valid Palindrome

Input: s = "A man, a plan, a canal: Panama"
Output: true
Explanation: "amanaplanacanalpanama" is a palindrome.

'F' - 'A' = 'f' - 'a'

CH - 'A' = ch - 'a'

ch = CH - 'A' + 'a'

s → A_man, a plan, a canal : Panama

↓

AmanaplanacanalPanama

↓

amanaplanacanalpanama

i ↑ ↑ j

```java
class Solution {
    public boolean isPalindrome(String s) {
        String t = "";

        for(int i = 0; i < s.length(); i++){
            char ch = s.charAt(i);

            if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z') || (ch >= '0' && ch <= '9')){
                if(ch >= 'A' && ch <= 'Z'){
                    // convert into smallcase / lowercase
                    ch = (char)(ch - 'A' + 'a');
                }
                t += ch;
            }

        }

        //question check if t string is palindrome or not
        int i = 0;
        int j = t.length()-1;

        while(i < j){
            if(t.charAt(i) != t.charAt(j)){
                return false;
            }
            i++;
            j--;
        }
        return true;

    }
}
```