Locate the Target String

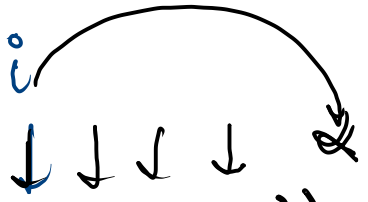
str = "geekster";

target = "st";

approch 1 →

g	e	e	k	s
ge	ee	ek	ks	st
gee	eek	eks	kst	
geek	eeks	ekst	kste	
geeks	eekst	ekste	kster	
geekst	eekste	ekster		
geekste	eekster			
geekster				

2 pointers



str = "geeksteer";

target = "ster";

we are only moving forward when char. are matching

for each 'i' , 'j' will start from 0

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

    locateTheTarget(str, target);
}

public static void locateTheTarget(String str, String tar) {
    for (int i = 0; i <= str.length() - tar.length(); i++) {
        for (int j = 0; j < tar.length(); j++) {
            if ( tar.charAt(j) != str.charAt(i + j) ) {
                break;
            }
            if (j == tar.length() - 1) {
                System.out.println(i);
                return;
            }
        }
    }
    System.out.println("-1");
    return;
}
```

str = "geeksterst";

tar = "st";

i=0, j=0 (s != g) T

i=1, j=0 (s != e) T

i=2, j=0 (s != e) T

i=3, j=0 (s != K) T

i=4, j=0 (s != S) False

j=1 (t != t) false

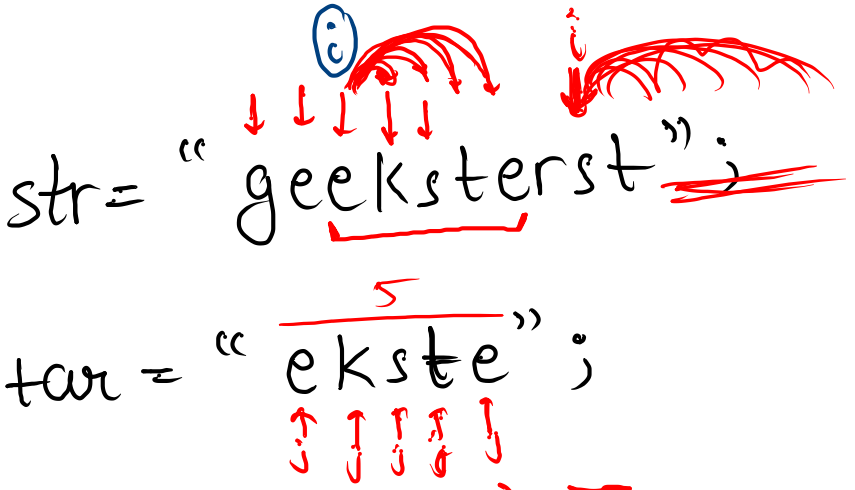
ans = 4

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

    locateTheTarget(str, target);
}

public static void locateTheTarget(String str, String tar) {
    for (int i = 0; i <= str.length() - tar.length(); i++) {
        for (int j = 0; j < tar.length(); j++) {
            if ( tar.charAt(j) != str.charAt(i + j) ) {
                break;
            }
            if (j == tar.length() - 1) {
                System.out.println(i);
                return;
            }
        }
    }
    System.out.println("-1");
    return;
}
```

// if j reached end of tar



i=0, j=0 (e != g) True

i=1, j=0 (e != e) False

j=1 (K != e) True

i=2, j=0 (e != e) False

j=1 (K != K) False

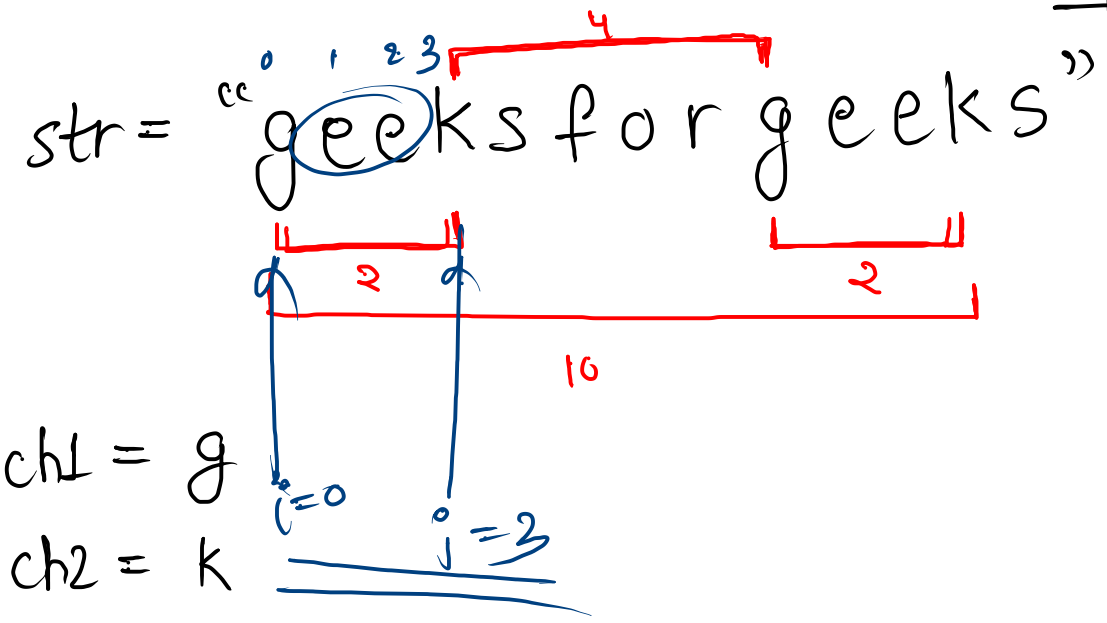
j=2 (s != s) False

j=3 (t != t) False

j=4 (e != e) False

ans = 2

Find Distance B/W Two Characters (2 pointers)



if we have to find diff. b/w 2 pointers (excluding both) then diff is (j - i - 1)

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    char ch1 = scn.next().charAt(0);
    char ch2 = scn.next().charAt(0);

    System.out.println(findDiff(str, ch1, ch2));
}

public static int findDiff(String str, char ch1, char ch2) {
    int ans = Integer.MAX_VALUE;
    for (int i = 0; i < str.length(); i++) {
        if (ch1 == str.charAt(i)) {
            for (int j = i; j < str.length(); j++) {
                if (ch2 == str.charAt(j)) {
                    ans = Math.min( ans, j - i - 1 );
                }
            }
        }
    }
    return ans;
}
```

str = geeksforgeeks

i ↓

j ↑

ch1 = g

ch2 = k

ans = ~~∞~~ 2

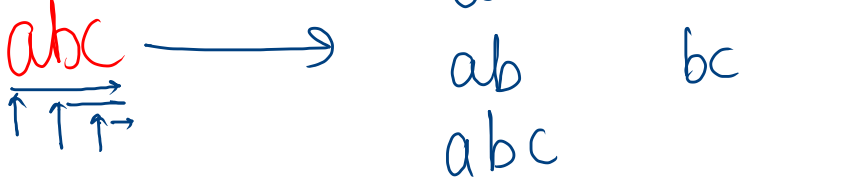
V. Imp

T.C O(N) , N = length of str

Substring ⇒ Subarray

(continuous part of string present in orginal one)

Print All Substrings



code

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

    for (int i = 0; i < str.length(); i++) {
        for (int j = i; j < str.length(); j++) {
            System.out.println( str.substring(i, j + 1) );
        }
    }
}
```

0 1 2 3 4 5 6 7

str = "geeksker"

excluded

1) str.substring(i, j+1);

accepts 2 parameter (si, ei+1), also accepts 1 para, si to end

str.substring(1, 3) → "ee"

str.substring(0, 5) → "geeks"

str.substring(5, 7) → "te"

str.substring(3, 8) → "kster"

2) str.substring(i)

str.substring(4, 9) →

// out of bound

str.substring(3) →

"kster"

str.substring(1) → "eekster"