Find Distance B/W Two Characters

Ex1 str = "Geeks"

0 1 2 3 4

ans = 2

ch1 = 'G' → (0) idx1

ans = idx2 - idx1 - 1;

ch2 = 'k' → (3) idx2

10

3 3

Ex2 str = "geeksfrgeeks"

0 1 2 3 4 5 6 7 8 9 10 11

↑ ch1 ↑ ch2

ch1 = 'g'

ch2 = 's'

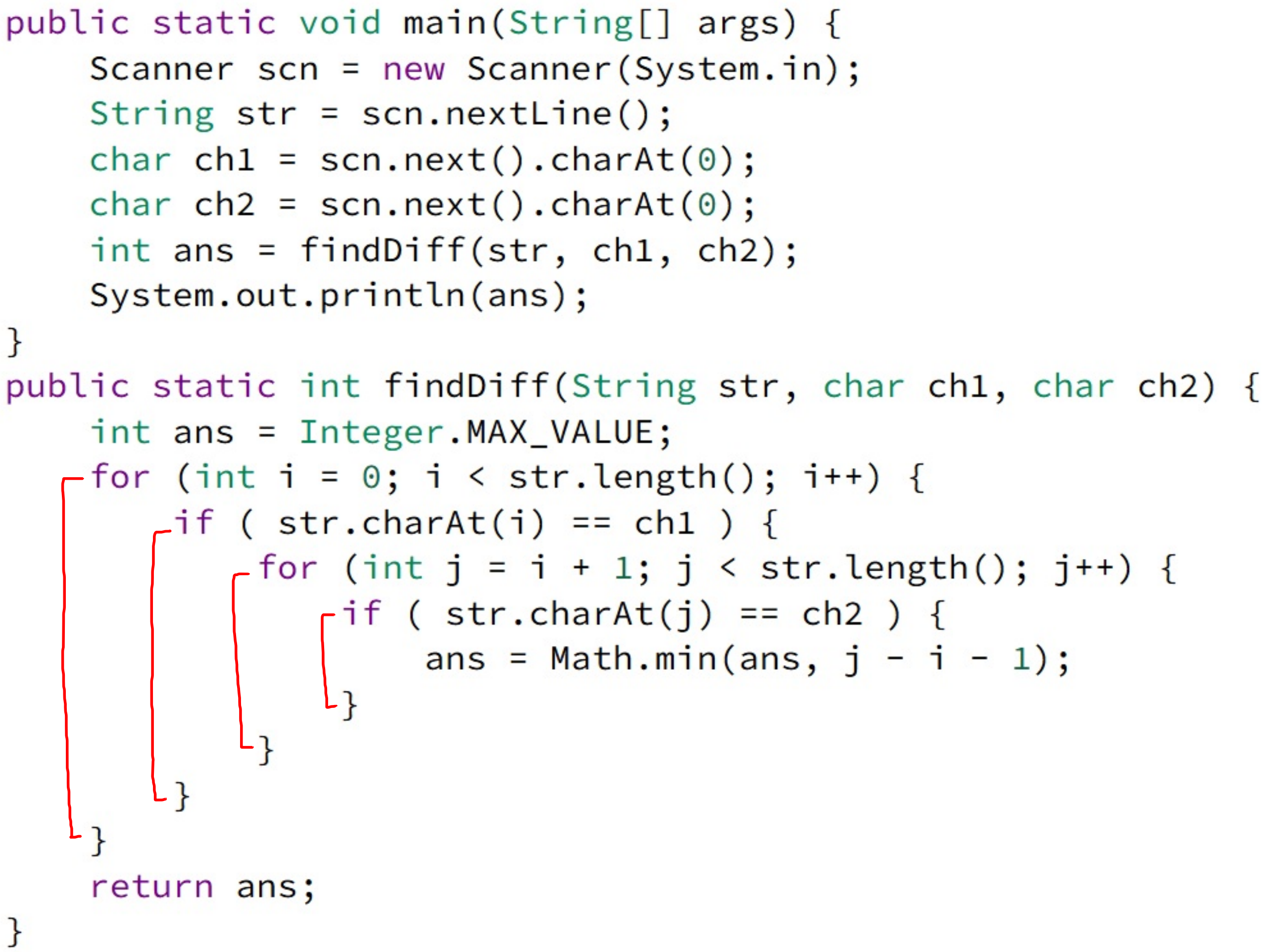
ans = 3

CwtR

psudo code

```
1) iterate a loop from to end
    1.1) check if we found ch1 → (i)
        1.1.1) iterate a loop from to end
            1.1.1.1) check if we found ch2 → (j)
                then diff = j - i - 1;
```

Code



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

Locate the Target String (V.Imp)

str = "geekstarsta"

0 1 2 3 4 5 6 7 8 9 10

↑

target = "sta"

0 1 2

ans = 4

brute force

- generate all substrings
- check if any of it is equal to target

$T.C = O(2^n) \longrightarrow O(n)$

issue

i

str = "geekstarsta"

g	e	e	k	s	t	a	r	s	t	a
0	1	2	3	4	5	6	7	8	9	10

target = "sta"

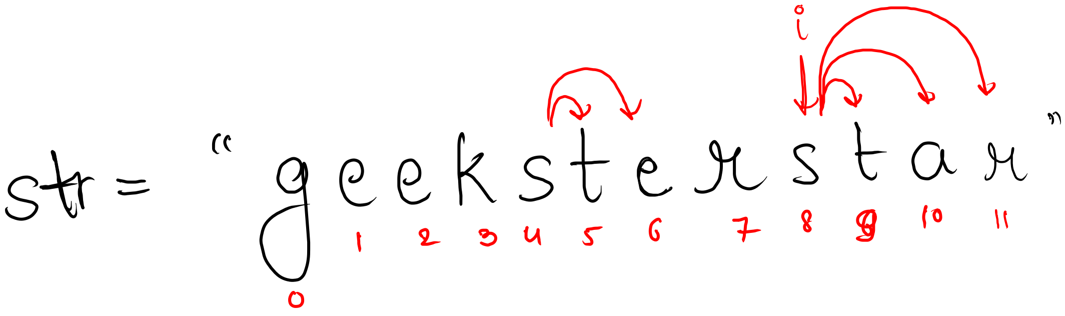
s	t	a
0	1	2

j

(charat $\underline{\underline{i+j}}$ == char at $\underline{\underline{\overset{6}{j}}}$)

't' 't'

'a' 'a'


str = "geeksterstar"

0 1 2 3 4 5 6 7 8 9 10 11

i

$i+j$

target = "star"

0 1 2 3

↑
j

i=4, j=0

j=1, t==t

j=2, a==e ×

i=8, j=0

j=1 t==t

j=2 a==a

j=3 r==r return i

psudo code

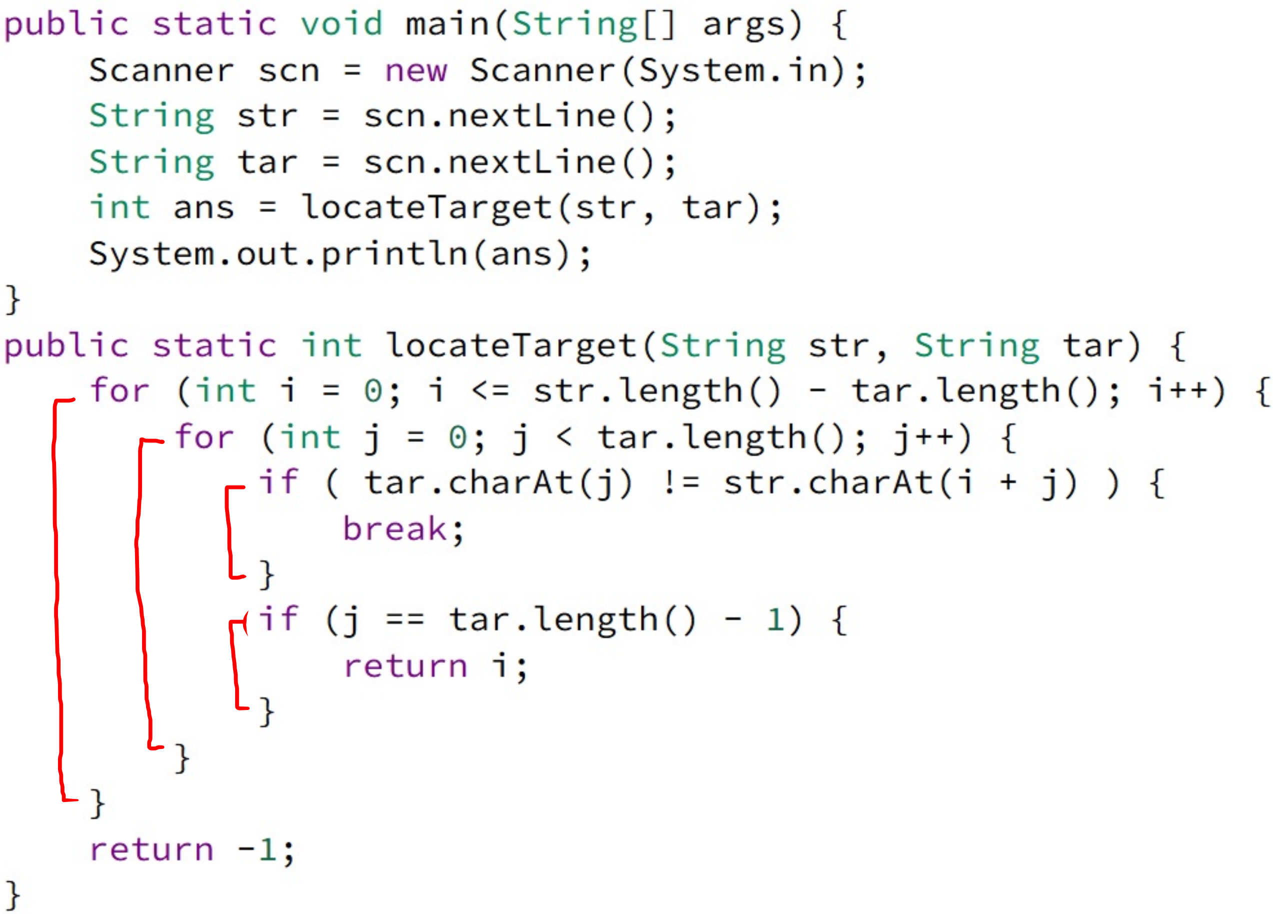
```
1) make 2 pointers
2) loop until i < n
    2.1) check charAt(j) in
         taget with charAt(i+j)
         in str is unequal
         break;
    2.2) if ( j == tar.len - 1)
             return i

return -1;
```

code

T.C = O(str.len * tar.len)

≈ O(N)



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    String tar = scn.nextLine();
    int ans = locateTarget(str, tar);
    System.out.println(ans);
}
public static int locateTarget(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) {
                return i;
            }
        }
    }
    return -1;
}
```

i
↓

str = "geeksterstar"

0 1 2 3 4 5 6 7 8 9 10 11

target = "star"

0 1 2 3

↑
j

S != g

S != e

S != e

S != K

S != S × → t != t ×

a != e ✓

S != t

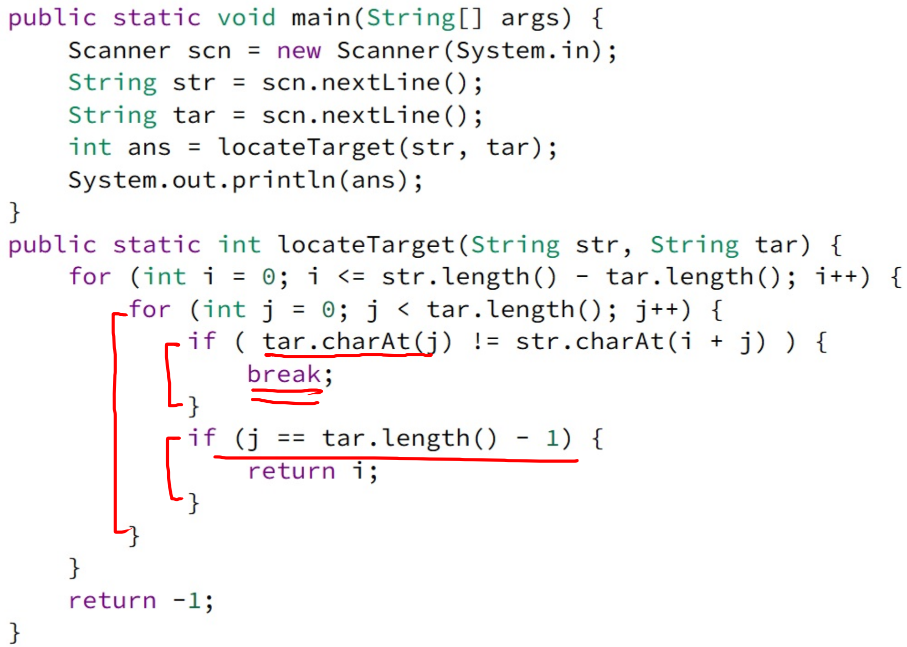
S != e

S != r

S != S × → t != t

a != a

r != r



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    String tar = scn.nextLine();
    int ans = locateTarget(str, tar);
    System.out.println(ans);
}
public static int locateTarget(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) {
                return i;
            }
        }
    }
    return -1;
}
```

⇒ Substring

inbuilt

```
str = "abcd"
       0123
```

syntex

```
str.substring(stard, end+1);

str.substring (1, 3);   // bc
str.substring (2);      // cd
```

Ques Generate all possible substrings

str = "abcd" ; 4   n = 4

0 1 2 3

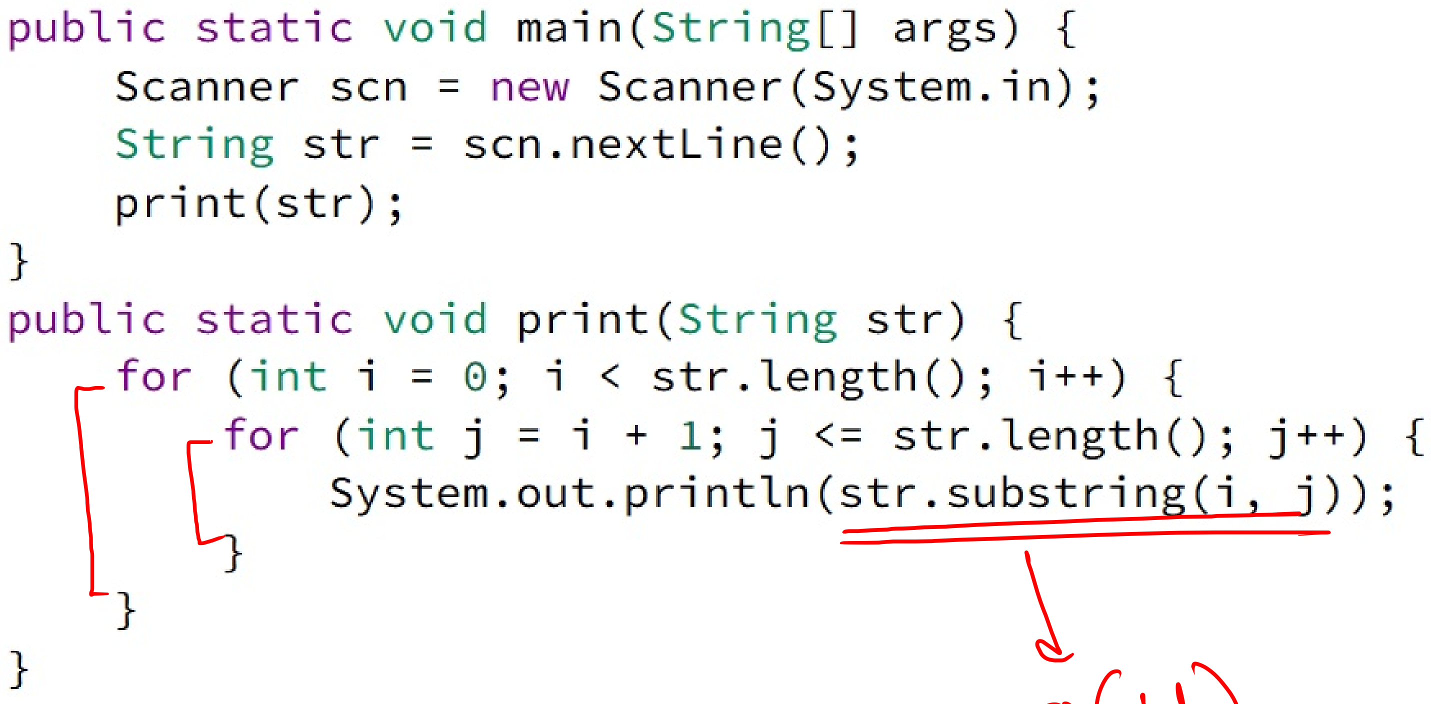
all substring

	i j
a	(0, 1)
ab	(0, 2)
abc	(0, 3)
abcd	(0, 4)
b	(1, 2)
bc	(1, 3)
bcd	(1, 4)
c	(2, 3)
cd	(2, 4)
d	(3, 4)

Code

```
for (int i=0; i<n; i++){
    for (int j=i+1; j<=n; j++){
        print str.substring(i,j);
    }
}
```

code



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    String str = scn.nextLine();
    print(str);
}
public static void print(String str) {
    for (int i = 0; i < str.length(); i++) {
        for (int j = i + 1; j <= str.length(); j++) {
            System.out.println(str.substring(i, j));
        }
    }
}
```

O(N)

N = str.length

O(N*N*N)