

1 2 3 4

①

1 2

1 2 3

1 2 3 4

$$\text{sum} = 13 + 123$$

12

int n = Integer.parseInt()

sum += n;

$$\text{sum} = 50 + 1 = 1 + 12 = 13$$

sum → Answer ✓

2

2 3

2 3 4

3

3 4

4

int sum = 0 {

int n = Integer.parseInt()

sum += n;

}

System.out.println(sum);

00110011

Sample Output 0

6

else {

grouped

00 11 → 2

00 11

01
00 11 } → 2

0011 → 2

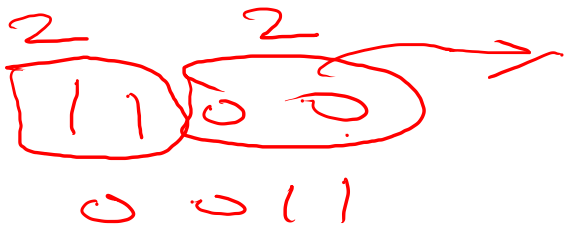
1100 → 2

ungrouped

0101
~~1111~~

0011 → 2

1100 → 2



$$ans = 0$$

$$ans += \min$$

$$ans = 2$$

00110011

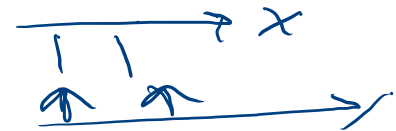


$$\begin{aligned} \text{Count zero} &= 0 \\ \text{Count one} &= 2 \end{aligned}$$

$$\min(0, 2)$$

$$0$$

$$ans = 4 + 2 = 6 + 0$$



$$6$$

When we are creating any type of array so we have always use the length attribute.

int[] arr = new int[n];

str.length() kavshik

(1)

class String {

public static int length (String str) {

int len = 0;

for (int i = 0; i < str.length(); i++) {

if (Character.isWhitespace()) {

len++;

}

return len;

}

0 0 1 0 1 →

1 0 0 1 1 0

return (base)

✓
adca
bdcq → X

adca
adca → X

adca → ✓
adca → ✓
adca → ✓

char[3] (ch1) = _____

(ch2)

1
1

abccba
cbaabc

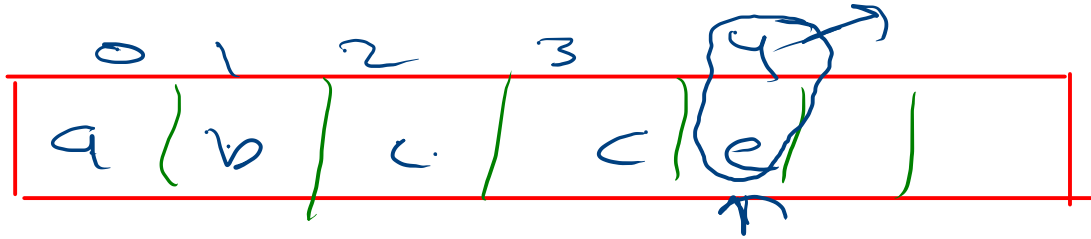
O(n)

a a b b c c
↓ ↓ ↓ ↓ ↓ ↓
a a b b c c

a a b b c c
c a a b b d d

hello world

2 →



e
101 - 97

4

4 + 97

101

h - 1

e - 1

l - 2

0 - 1
1 - 2

w - 1

r - 1

d - 1



-1 > 0

~~0~~

-1 > 0

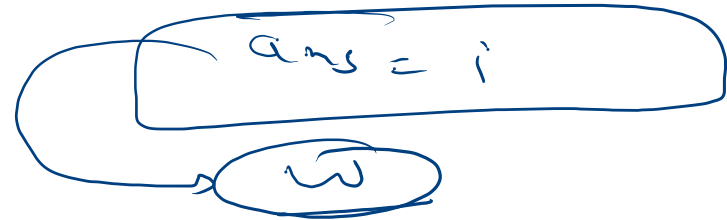
-1 > 0

-1 > 1

$k = 2$

max = 1 1 > 1

if (1 > 1) 1 > 1



HW Check Anagram

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Sol
11         Scanner sc = new Scanner(System.in);
12         String str1 = sc.next();
13         String str2 = sc.next();
14
15         boolean ans = isAnagram(str1,str2);
16         if(ans){
17             System.out.println("True");
18         }else{
19             System.out.println("False");
20         }
21     }
22 }
23 public static boolean isAnagram(String str1,String str2){
24     if(str1.length()!=str2.length()){
25         return false;
26     }
27     int[] freq = new int[26];
28     for(int i=0;i<str1.length();i++){
29         char ch = str1.charAt(i);
30         int cal = (int)(ch-97);
31         freq[cal]+=1;
32     }
33
34     // empty
35     for(int i=0;i<str2.length();i++){
36         char ch = str2.charAt(i);
37         int cal = (int)(ch-97);
38         freq[cal]-=1;
39     }
40     // checking
41     for(int i=0;i<26;i++){
42         if(freq[i]>0){
43             return false;
44         }
45     }
46     return true;
47 }
48 }
```

HW_K Frequent Characters

Language: Java 7

 Open in editor

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11         Scanner sc = new Scanner(System.in);
12         String str = sc.nextLine();
13         int k = sc.nextInt();
14
15         frequent(str,k);
16     }
17     public static void frequent(String str,int k){
18         str = str.replaceAll(" ", "");
19         int[] freq = new int[26];
20         for(int i=0;i<str.length();i++){
21             char ch = str.charAt(i);
22             int cal = (int)(ch-97);
23             freq[cal]+=1;
24         }
25         // int count=0;
26         while(k>0){
27             int max=-1;
28             int ans=-1;
29             for(int i=0;i<26;i++){
30                 if(freq[i]>max){
31                     max=freq[i];
32                     ans=i;
33                 }
34             }
35             freq[ans]=-1;
36             k--;
37             System.out.print((char)(ans+97)+" ");
38         }
39     }
40 }
```