Daily Challenge   **Happy Coding from necse**   ☑ **SkillRack**

Digits Asterisks Pattern

The program must accept a string **S** containing only digits as the input. The program must print the digits in the string S based on the following conditions.
- The digits of the same value must be printed on the same line.
- The values of the digits must be printed in the order of their occurrence.
- The empty spaces in each row must be printed as asterisks.

Boundary Condition(s):
1 <= Length of S <= 100

Input Format:
The first line contains S.

Output Format:
The lines contain the digits and asterisks based on the given conditions.

Example Input/Output 1:
Input:
1225644789964

Output:
1 * *
2 2 *
5 * *
6 6 *
4 4 4
7 * *
8 * *
9 9 *

Example:
Here S = **1225644789964**.
1 occurs one time.
2 occurs two times.
5 occurs one time.
6 occurs two times.
4 occurs three times.
7 occurs one time.
8 occurs one time.
9 occurs two times.

Example Input/Output 2:
Input:
624655622653

Output:
6 6 6 6
2 2 2 *
4 * * *
5 5 5 *
3 * * *

Example Input/Output 3:
Input:
1234567

Output:
1
2
3
4
5
6
7

Max Execution Time Limit: 50 millisecs

Ambiance

```java
1  import java.util.*;
2  public class Hello {
3
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          Map<Integer,Integer> ht = new LinkedHashMap<>();
8          int max=1;
9
10         char arr[] = sc.nextLine().toCharArray();
11
12
13         for(char i:arr){
14             if(ht.containsKey(i-'0')){
15                 ht.put(i-'0',ht.get(i-'0')+1);
16                 max=Math.max(ht.get(i-'0'),max);
17             }else{
18                 ht.put(i-'0',1);
19             }
20         }
21
22         for(Map.Entry<Integer,Integer> e:ht.entrySet()){
23
24             for(int i=0;i<e.getValue();i++){
25                 System.out.print(e.getKey()+" ");
26             }
27
28             for(int i=0;i<max-e.getValue();i++){
29                 System.out.print("* ");
30             }
31             System.out.println("");
32         }
33
34
35     }
36 }
```

1912067@nec

Code did not pass the execution   — ✕

Input:

1225644789964

Expected Output:

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

Your Program Output:

```
1 * * -
2 2 * -
5 * * -
6 6 * -
4 4 4 -
7 * * -
8 * * -
9 9 * -
```

Save Run