Name : Aakash Kumar Pandey
Redg.No : 2041016071

Program logic Description:

In the given program, we are enter a large number (n>0).Now we are taking large number as a string. Then we are calculating the length of a string. After that we are taking the loop and initializing it 0 and repeating till length-1 and updating the counter. Then we are using charAt method to find position of each character in the loop. Then we are taking another loop and updating string with string character position . Then we are taking another loop if (n == numb.charAt(j) && numb.charAt(j) != '*') we are printing star '*' because repetition of number is taking place . Else we are updating string we that number of the position using str += numb.charAt(j);. After that we are updating string and printing each unique value
in the given number.

Then, in the second part of our Question we have to print largest number using unique number we obtain from our previous result. So, calculating length of unique number obtain and after that we are taking loop and taking empty variable. Then we take another loop in that if(numb.charAt(i) != '#' than using parseInt method for converting string to integer "if (max < Integer.parseInt("" + numb.charAt(i)))" then updating max by Integer.parseInt("" + numb.charAt(i)) after that we check for each term and updating maximum. By this numb = numb.replace("" + max, "#"); we place first maximum by # and we go for second maximum and we do that each term till length-1. Then we print maximum by System.out.println((j + 1) + " iterartion lrg:" + lrg + " str:" + numb);

Output:

Test case 1:

```
Enter a large number: 12134616235835
1 iteration "12*346*6235835"
2 iteration "12*346*6*35835"
3 iteration "12*346*6*35835"
4 iteration "12*346*6**58*5"
5 iteration "12*346*6**58*5"
6 iteration "12*346****58*5"
7 iteration "12*346****58*5"
8 iteration "12*346****58*5"
9 iteration "12*346****58*5"
10 iteration "12*346****58*5"
11 iteration "12*346****58**"
12 iteration "12*346****58**"
13 iteration "12*346****58**"
14 iteration "12*346****58**"
1 iterartion   lrg:8        str:123465#
2 iterartion   lrg:86       str:1234#5#
3 iterartion   lrg:865      str:1234###
4 iterartion   lrg:8654     str:123####
5 iterartion   lrg:86543    str:12#####
6 iterartion   lrg:865432   str:1######
```

7 iterartion lrg:8654321 str:#######

The unique digits present in 12134616235835 are 1, 2, 3, 4, 6, 5, 8. The largest number possible out of these unique digits is 8654321.

Test case 2:
Enter a large number: 11131116111811
1 iteration "1**3***6***8**"
2 iteration "1**3***6***8**"
3 iteration "1**3***6***8**"
4 iteration "1**3***6***8**"
5 iteration "1**3***6***8**"
6 iteration "1**3***6***8**"
7 iteration "1**3***6***8**"
8 iteration "1**3***6***8**"
9 iteration "1**3***6***8**"
10 iteration "1**3***6***8**"
11 iteration "1**3***6***8**"
12 iteration "1**3***6***8**"
13 iteration "1**3***6***8**"
14 iteration "1**3***6***8**"
1 iterartion lrg:8 str:136#
2 iterartion lrg:86 str:13##
3 iterartion lrg:863 str:1###
4 iterartion lrg:8631 str:####

The unique digits present in 11131116111811 are 1, 3, 6, 8. The largest number possible out of these unique digits is 8631.
Test case 3:
Enter a large number: 7
1 iteration "7"
1 iterartion lrg:7 str:#
The unique digits present in 7 is 7. The largest number possible out of these unique digits is 7.

Test case 4:
Enter a large number: 11111111111
1 iteration "1**********"
2 iteration "1**********"
3 iteration "1**********"
4 iteration "1**********"
5 iteration "1**********"
6 iteration "1**********"
7 iteration "1**********"

8 iteration "1**********"
9 iteration "1**********"
10 iteration "1**********"
11 iteration "1**********"
1 iterartion lrg:1 str:#

The unique digits present in 11111111111 are 1. The largest number possible out of these unique digits is 1.

Test case 5:
Enter a large number: 1253478690
1 iteration "1253478690"
2 iteration "1253478690"
3 iteration "1253478690"
4 iteration "1253478690"
5 iteration "1253478690"
6 iteration "1253478690"
7 iteration "1253478690"
8 iteration "1253478690"
9 iteration "1253478690"
10 iteration "1253478690"
1 iterartion lrg:9 str:12534786#0
2 iterartion lrg:98 str:125347#6#0
3 iterartion lrg:987 str:12534##6#0
4 iterartion lrg:9876 str:12534####0
5 iterartion lrg:98765 str:12#34####0
6 iterartion lrg:987654 str:12#3#####0
7 iterartion lrg:9876543 str:12#######0
8 iterartion lrg:98765432 str:1########0
9 iterartion lrg:987654321 str:#########0
10 iterartion lrg:9876543210 str:##########

The unique digits present in 1253478690 are 1,2,5,3,4,7,8,6,9,0. The largest number possible out of these unique digits is 9876543210.

Test case 6:
Enter a large number: 000000000
1 iteration "0********"
2 iteration "0********"
3 iteration "0********"
4 iteration "0********"
5 iteration "0********"
6 iteration "0********"
7 iteration "0********"
8 iteration "0********"
9 iteration "0********"
1 iterartion lrg:0 str:#

The unique digits present in 000000000 is 0. The largest number possible out of these unique digits is 0.

Test case 7:
Enter a large number: 1222222222
1 iteration "1222222222"
2 iteration "12********"
3 iteration "12********"
4 iteration "12********"
5 iteration "12********"
6 iteration "12********"
7 iteration "12********"
8 iteration "12********"
9 iteration "12********"
10 iteration "12********"
1 iterartion lrg:2 str:1#
2 iterartion lrg:21 str:##

The unique digits present in 1222222222 are 1,2. The largest number possible out of these unique digits is 21.
Test case 8:
Enter a large number: 33333333335
1 iteration "3*********5"
2 iteration "3*********5"
3 iteration "3*********5"
4 iteration "3*********5"
5 iteration "3*********5"
6 iteration "3*********5"
7 iteration "3*********5"
8 iteration "3*********5"
9 iteration "3*********5"
10 iteration "3*********5"
11 iteration "3*********5"
1 iterartion lrg:5 str:3#
2 iterartion lrg:53 str:##

The unique digits present in 33333333335 are 3,5. The largest number possible out of these unique digits is 53.