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

Integer Compression - Right to Left

The program must accept an integer **N** as the input. The program must compress the integer by concatenating the sum of every two digits from right to left in N until the integer becomes a single-digit integer. The program must print the integer values obtained during the compression process(including N) as the output.

Boundary Condition(s):
0 <= N <= 10^8

Input Format:
The first line contains N.

Output Format:
The lines contain the integer values based on the given conditions.

Example Input/Output 1:
Input:
2345677

Output:
2345677
271114
925
97
16
7

Explanation:
Here N = 2345677.
2345677 -> (2) (3+4) (5+6) (7+7) -> 2 7 11 14
271114 -> (2+7) (1+1) (1+4) -> 9 2 5
925 -> (9) (2+5) -> 9 7
97 -> (9+7) -> 16
16 -> (1+6) -> **7**

Example Input/Output 2:
Input:
1234566

Output:
1234566
15912
1143
27
9

Explanation:
Here N = 1234566.
1234566 -> (1) (2+3) (4+5) (6+6) -> 1 5 9 12
15912 -> (1) (5+9) (1+2) -> 1 14 3
1143 -> (1+1) (4+3) -> 2 7
27 -> (2+7) -> **9**

Max Execution Time Limit: 50 millisecs

Ambiance

Java (12.0)

```java
import java.util.*;
public class Hello {


    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        String s = sc.nextLine();
        System.out.println(s);
        solveNumber(new StringBuilder(s).reverse().toString());
        System.out.println("");
    }

    public static void solveNumber(String s){
        if(s.length()==1){
            return;
        }
        String res = "";
        int i=0;
        for(i=0;i<s.length()-1;i+=2){
            res = ((s.charAt(i)-'0')+
                  (s.charAt(i+1)-'0'))+res;
        }

        if(i!=s.length()){
            res = (s.charAt(s.length()-1)-'0')+res;
        }

        System.out.println(res);
        solveNumber(new StringBuilder(res).reverse().toString());

    }
}
```

1912067@nec

Code did not pass the execution

Input:

```
2345677
```

Expected Output:

```
2345677
271114
925
97
16
7
```

Your Program Output:

```
2345677
271114
925
97
16
7
-
```

Save Run