6. Forming New Word from a String

Write a program to read a string and a positive integer n as input and construct a string with first n and last n characters in the given string.
Include a class **UserMainCode** with a static method **formNewWord** which accepts a string
and positive integer .

The return type of the output should be a string (value) of first n character and last n character.
Create a class **Main** which would get the input as a string and integer n and call the static method**formNewWord** present in the UserMainCode.
Input and Output Format:
Input consists of a string of even
length. Output is a string .
Note: The given string length must be >=2n.
Refer sample output for formatting specifications.
Sample Input 1:
California
3
Sample Output 1:
Calnia
Sample Input 2:
this
1
Sample Output 2:
Ts

MAIN:

```
=====
Import java.util.*;
Public class Main {
Public static void main (String [] args)
{

}}
```

USERMAINCODE:

```
==============
public class UserMainCode {
public static String formNewWord (String s1, int n) {

}}
```

7. Reversing a Number

Write a program to read a positive number as input and to get the reverse of the given number and return it as output.
Include a class **UserMainCode** with a static method **reverseNumber** which accepts a positive
integer .
The return type is an integer value which is the reverse of the given number. Create a **Main** class which gets the input as a integer and call the static method **reverseNumber** present in the **UserMainCode**
Input and Output Format: Input
consists of a positive integer.
Output is an integer .
Refer sample output for formatting specifications.
Sample Input 1:
543
Sample Output 1:
345
Sample Input 1:
1111
Sample Output 1:
1111

MAIN:

```
=====
Import java.util.*;
Public class Main {
Public static void main (String [] args)
{

}}
```

USERMAINCODE:

```
==============
public class UserMainCode {
public static int reverseNumber(int n){

}

}
```

8. ArrayList Sorting and Merging

Write a code to read two int array lists of size 5 each as input and to merge the two arrayLists, sort the merged arraylist in ascending order and fetch the elements at 2nd, 6th and 8th index into a new arrayList and return the final ArrayList.
Include a class **UserMainCode** with a static method **sortMergedArrayList** which accepts 2
ArrayLists.
The return type is an ArrayList with elements from 2,6 and 8th index position .Array index starts from position 0.
Create a **Main** class which gets two array list of size 5 as input and call the static method**sortMergedArrayList** present in the **UserMainCode.**
Input and Output Format:
Input consists of two array lists of
size 5. Output is an array list .
Note - The first element is at index 0.
Refer sample output for formatting specifications.

Sample Input 1:

```
3
1
17
11
19
5
2
7
6
20
```

Sample Output 1:

```
3
11
19
```

Sample Input 2:
1
2
3
4
5
6
7
8
9
10
Sample Output 2:
3
7
9

MAIN:

```
=====
Import java.util.*;
Public class Main {
Public static void main (String [] args)
{

}}
```

USERMAINCODE:

```
==============
public class UserMainCode {
public static ArrayList<Integer> sortMergedArraylist (ArrayList<Integer>
                                                    list1, ArrayList<Integer>list2) {

}}
```

9. Password Validation

Given a method with a password in string format as input. Write code to validate the password using following rules:
- Must contain at least one digit
- Must contain at least one of the following special characters @, #, $ # Length should be between 6 to 20 characters.

Include a class **UserMainCode** with a static method **validatePassword** which accepts a password string as input.
If the password is as per the given rules return 1 else return -1.If the return value is 1 then print valid password else print as invalid password.
Create a **Main** class which gets string as an input and call the static method **validatePassword** present in the **UserMainCode.**

Input and Output Format:
Input is a string .
Output is a string . **Sample Input 1:**
%Dhoom%
Sample Output 1:
Invalid password
Sample Input 2:
#@6Don
Sample Output 2:
Valid password

MAIN:

```
=====
Import java.util.*;
Public class Main {
Public static void main (String [] args)
{

}}
```

USERMAINCODE:

```
==============
public class UserMainCode {
public static int validatePassword (String password){

}}
```

10. Sum of Powers of elements in an array

Given a method with an int array. Write code to find the power of each individual element accoding to its position index, add them up and return as output.
Include a class **UserMainCode** with a static method **getSumOfPower** which accepts an integer array as input.
The return type of the output is an integer which is the sum powers of each element in the array.
Create a **Main** class which gets integer array as an input and call the static method **getSumOfPower**present in the **UserMainCode.**
Input and Output Format:
Input is an integer array.First element corresponds to the number(n) of elements in an array.The next inputs corresponds to each element in an array.
Output is an integer .
Sample Input 1:
4
3
6
2
1
Sample Output 1:
12

Sample Input 2:
4
5
3
7
2
Sample Output 2:
61

MAIN:
=====

```
Import java.util.*;
Public class Main {
Public static void main (String [] args)
{

}}
```

USERMAINCODE:

```
==============
public class UserMainCode {
public static int getSumOfPower(int n ,int[] a)
}}
```