Practice Problem #1

- Write a program that has the class name `Problem1` and that has the `main` method.

- In the main method prompt the user to enter positive number up to 20 (inclusive) for up to 5 times. It then counts and adds the positive numbers.

- Your prompt terminates whenever the user enters 5 numbers or non-positive number. You can assume that there is at-least one non zero number.

- Print out the count of the positive numbers and their product.

- Several sample runs are provided for you below. Your output must be formatted **exactly** like the sample runs below.

```
Enter a positive integer: 12
Enter a positive integer: 17
Enter a positive integer: 2
Enter a positive integer: 0
Number of positive numbers is: 3
The product of positive numbers is: 408
```

```
Enter a positive integer: 6
Enter a positive integer: 5
Enter a positive integer: 7
Enter a positive integer: 3
Enter a positive integer: 19
Number of positive numbers is: 5
The product of positive numbers is: 11970
```

```
Enter a positive integer: 356
Enter a positive integer: -6
Number of positive numbers is: 1
The product of positive numbers is: 356
```

Practice Problem #2

- This is a Follow up question from `Problem` 1. Write a method named `SumFromProduct` that takes an integer and returns an integer.

- The method parameter is a product that you get from `Problem` 1. The method then calculates the sum of the digits in the product and returns the value.

- If the sum is greater than 10 (exclusive) then reduce the sum again to value between 1 and 10 (exclusive) by adding the individual digits.

- Several sample runs are provided for you below. Your output must be formatted **exactly** like the sample runs below.

Sample Method Usage	return
`SumFromProduct(408)`	3
`SumFromProduct(11970)`	9
`SumFromProduct(356)`	5

Practice Problem #3

- This is a follow up from `Problem 2` and `Problem 2`. Write a method named `SumFactorial` that takes an integer and returns an integer.

- The input value is the sum that you get from `Problem 2`. The method then calculates and returns the factorial of the sum.

- The factorial of a number `n` (called `n` factorial or `n!`) is the product of the integers from 1 up to and including `n`.

- Several sample runs are provided for you below. Your output must be formatted **exactly** like the sample runs below.

Sample Method Usage	return
SumFactorial(3)	6
SumFactorial(9)	362880
SumFactorial(5)	120

- You should call `SumFromProduct` and `SumFactorial` from the main method in `Problem1` only. If you implemented your methods correctly, the output should match the following:

```
Enter a positive integer: 12
Enter a positive integer: 17
Enter a positive integer: 2
Enter a positive integer: 0
Number of positive numbers is: 3
The product of positive numbers is: 408
The sum of the product is: 3
The factorial of 3 is: 6
```

```
Enter a positive integer: 6
Enter a positive integer: 5
Enter a positive integer: 7
Enter a positive integer: 3
Enter a positive integer: 19
Number of positive numbers is: 5
The product of positive numbers is: 11970
The sum of the product is: 9
The factorial of 9 is: 362880
```

```
Enter a positive integer: 356
Enter a positive integer: -6
Number of positive numbers is: 1
The product of positive numbers is: 356
The sum of the product is: 5
The factorial of 5 is: 120
```