

## Week 4

Write and run your programs with IDLE editor. Submit finished programs to CodeGrade. Note that some tasks have several steps (A, B, C, ...) in CodeGrade.

**IMPORTANT:** End each input-command string with a newline symbol `\n`. For example:

```
variable = input("Some text:\n")
```

**Task 1:** The **factorial** of a non-negative integer  $n$  is the product of all positive integers less than or equal to  $n$ . The factorial of  $n$  is denoted  $n!$ . For example,  $5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$ . In addition, the factorial of zero is defined  $0! = 1$ .

Write a program to calculate the factorial of a number using a `for` loop. If the input is negative, the program reports an error according to the example below.

### Example run 1:

```
Enter a non-negative integer:
5
Factorial of 5 is 120
```

### Example run 2:

```
Enter a non-negative integer:
0
Factorial of 0 is 1
```

### Example run 3:

```
Enter a non-negative integer:
-3
Error: Factorial is not defined for negative numbers
```

**Task 2:** A number is even if it is divisible by 2. Write a program that asks for a number and then prints all even numbers from 1 to the number. Note that the number itself is included if it is even. The numbers are separated by three dots.

### Example run 1:

```
Enter a positive integer:
20
2...4...6...8...10...12...14...16...18...20...
```

### Example run 2:

```
Enter a positive integer:
0
0 is not positive
```

**Task 3:** Commonly it is agreed that there are five vowels in the English language: **a, e, i, o, u**. Write a program to count the number of vowels in a given string. Note that both lower and upper case letters are considered vowels.

**Example run 1:**

```
Enter a string:
The USA, the shorthand for the United States of America.
Number of vowels is: 18
```

**Task 4:** Write a program that prints numbers based on specific rules. Initially, you'll ask the user to input two numbers, which we'll call "a" and "b". After that, you'll start printing these numbers in a loop. In each round of the loop, the value of "a" will double, and the value of "b" will increase by 100. The loop will stop when either "a" or "b" (or both) becomes greater than 1000. Your program should also tell the reason that caused the loop to end.

**Example run 1:**

```
Enter a:
5
Enter b:
333
a: 5 b: 333
a: 10 b: 433
a: 20 b: 533
a: 40 b: 633
a: 80 b: 733
a: 160 b: 833
a: 320 b: 933
b exceeded 1000
```

**Example run 2:**

```
Enter a:
2
Enter b:
100
a: 2 b: 100
a: 4 b: 200
a: 8 b: 300
a: 16 b: 400
a: 32 b: 500
a: 64 b: 600
a: 128 b: 700
a: 256 b: 800
a: 512 b: 900
a exceeded 1000
```

**Example run 3:**

```
Enter a:
2
Enter b:
111
a: 2 b: 111
a: 4 b: 211
a: 8 b: 311
a: 16 b: 411
a: 32 b: 511
a: 64 b: 611
a: 128 b: 711
a: 256 b: 811
a: 512 b: 911
a exceeded 1000
b exceeded 1000
```

**Task 5:** By using **for**-loop write a program that modifies a given string. Follow these steps:

1. Prompt the user to input a string.
2. Replace every occurrence of the letter "s" with "z". And every instance of "S" with "Z".
3. Display the modified string as the output.

Ensure that your program handles both lowercase and uppercase instances of the letter "s" and "S" appropriately.

**Example run 1:**

```
Enter a string:
This is a sample string with some 'S' characters.
Modified string: Thiz iz a zample ztring with zome 'Z' characterz.
```