# Object-Oriented Programming (3190)

# Homework 2

# Spring 2023

**1. [5pt]** Write a program that prints the quarter (1, 2, 3, and 4) of a point in the Cartesian (rectangular) system given the values of x and y for the point. For example, if both x and y are positive, the point is located in the first quadrant. If both x and y are negative, the point is located in the third quadrant, and so on. Make sure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Enter x location: 5
Enter y location: 7
The point (5 , 7) is in the first quartile.
```

```
Run:
Enter x location: -4
Enter y location: 8
The point (-4 , 8) is in the second quartile.
```

```
Run:
Enter x location: -7
Enter y location: -3
The point (-7 , -3) is in the third quartile.
```

```
Run:
Enter x location: 7
Enter y location: -4
The point (7 , -4) is in the fourth quartile.
```

**2. [5pt]** Write a program that reads a positive integer between 1 and 100 and prints its factors (divisors). A factor is a number that divides another number. For example, the factors of 10 are 1, 2, 5, and 10. The factors of 12 are 1, 2, 3, 4, 6, and 12. Ensure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Enter a number between 1 & 100: 78

Factors are:
1 2 3 6 13 26 39 78
```

**3. [5pt]** Define a class named Triangle as follows:

a. Data members are firstSide, secondSide, and thirdSide.

b. Use a constructor that asserts that the sum of any two sides to be greater than the third one.

c. Accessor member functions are getSides, getPerimeter, and getArea. To find the perimeter and area of a triangle, use the following:

```
perimeter = a + b + c
area = sqrt ((p) * (p - a) * (p - b) * (p - c)) // p = perimeter / 2
```

d. Implement the interface, implementation, and application files.

Make sure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Triangle1 (5, 4, 3)
Perimeter: 12
Area: 6.00

Triangle2 (6, 7, 9)
Perimeter: 22
Area: 20.98
```

**4. [5pt]** Write a function that, given two strings, creates another string that contains only the common characters of the two strings. One solution is to first remove the duplicate characters from each string before creating the string with common characters. Ensure that your output matches the following output and test your code with at least two more test cases on your own.

Possible Output:

```
Run:
Enter first string: AAABCBBCDDEEEFG
Enter second string: BCCCDFGG
String with common characters:
BCDFG
```

```
Run:
Enter first string: BBAACCDDDEFGA
Enter second string: AAABBCFF
String with common characters:
BACF
```

**End of Assignment.**