Refactoring and Functions

For this exercise, you will be provided with several different source files. You will need to edit these files to clean them up.

Exercise 1: Circles

Download **circles.cpp**. Add the following functions to the program and adjust main():

1. **CalculateCircumference**
   Return type: Float – the circumference
   Parameters: Float – Radius

   Calculate the circumference and return the value

2. **CalculatePerimeter**
   Return type: Float – The perieter
   Parameters: Float – Radius

   Calculate the perimeter and return the value

Exercise 2: Cipher

Download **cipher.cpp**. Add the following functions to the program and adjust main():

1. **OffsetCharacter**
   Return type: Character – the converted character
   Parameters: Character – the original character
                      Integer – the cipher key

   Within the function, change the value of the original character by adding the cipher key to its character code (ASCII code). Refactor the logic that is already there and pull it out into this function.

Exercise 3: Arithmetic

Download **arithmetic.cpp**. Add the following functions to the program and adjust main():

1. **MainMenu**
   Return type: Void
   Parameters: None

   Display the top numbered menu (1. Add, 2. Subtract, etc.)

2. **GetInput**
   Return type: Integer – The user's selection
   Parameters: Integer – minimum value
                      Integer – maximum value

   Have the user input a choice. As long as the choice is less than the minimum, or greater than the maximum, have them re-enter their choice. Return the choice once they've entered a valid #.

3. **AddTwo**
   Return type: Integer – The sum
   Parameters: None

   Within the function, ask the user to enter two numbers. Return the sum of these two.

4. **SubtractTwo**
   Return type: Integer – The difference
   Parameters: None

   Within the function, ask the user to enter two numbers. Return the difference of these two.

5. **MultiplyTwo**
   Return type: Integer – The product
   Parameters: None

   Within the function, ask the user to enter two numbers. Return the quotient of these two.

6. **DivideTwo**
   Return type: Integer – The quotient
   Parameters: None

   Within the function, ask the user to enter two numbers. Return the product of these two.