*TASK 2 – **rectangle.rb***

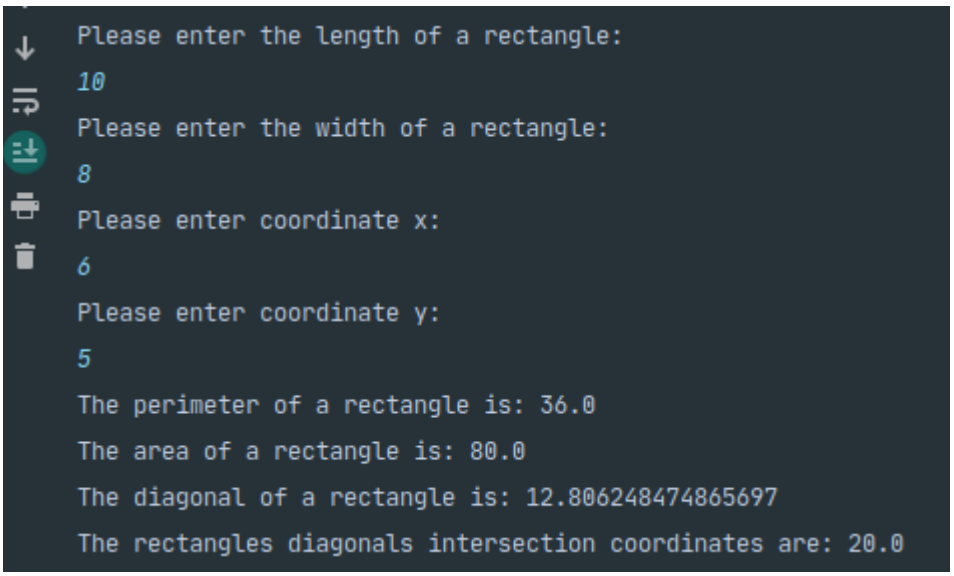
This task was chosen from this repository: https://github.com/dziugaspeciulevicius/Object-Oriented-Programming/tree/master/TASK1/src/com/company

This task required to calculate rectangle perimeter, square size, diagonal length and rectangle diagonals intersection coordinates.

These are the methods to calculate the required aspects:

```ruby
# method to get a perimeter of a rectangle
def get_perimeter(width, length)
  (2 * length) + (2 * width)
end

# method to get area of a rectangle
def get_area(width, length)
  width * length
end

# method to calculate rectangles diagonal length
def get_diagonal_length(width, length)
  Math.sqrt((length * length) + (width * width))
end

# method to get x coordinate
def get_coordinate_x(coordinate_x, length)
  coordinate_x + length / 2
end

# method to get y coordinate
def get_coordinate_y(coordinate_y, width)
  coordinate_y + width / 2
end
```

When running the program it requires to enter the length and the width of a rectangle and x,y coordinates. After entering these properties it will give us our calculation:



---

*TASK2 – **vigenere.rb***

This task was chosen from this repository:
https://github.com/dziugaspeciulevicius/Information-Security/blob/master/TASK1/src/com/company/Main.java

To complete this task I had to install a library for it to work.

```ruby
# vigenere cipher library (gem install vigenere)
require 'caesar'
```

After that I wrote encryption and decryption methods. Had to find some help from fellow programmers on the internet, because it was quite hard to get it to work on ruby.

```ruby
# encrypt function which takes a key and a plain text
def encrypt(key, plain_text)
  key = key.upcase.split('')
  cipher_text = plain_text.upcase.split('').collect do |letter|
    if !('A'..'Z').include?(letter)
      cipher_letter = letter
    else
      cipher_letter = Caesar.encode(key.first, letter)
      key << key.shift
    end
    cipher_letter
  end
  cipher_text.join
end

# decrypt function which takes a key and a cipher text
def decrypt(key, cipher_text)
  key = key.upcase.split('')
  plain_text = cipher_text.split('').collect do |cipher_letter|
    if !('A'..'Z').include?(cipher_letter)
      letter = cipher_letter
    else
      letter = Caesar.decode(key.first, cipher_letter)
      key << key.shift
    end
    letter
  end
  plain_text.join
end
```

```
Enter plain text to cipher:
My name is Dziugas
Enter key:
Hello
Original: My name is Dziugas
Encrypted: TC YLAL MD ONPYRLG
Decrypted: MY NAME IS DZIUGAS
```

```
Enter plain text to cipher:
I'm 22 years old
Enter key:
keey
Original: I'm 22 years old
Encrypted: S'Q 22 CCKVW MVH
Decrypted: I'M 22 YEARS OLD
```

The cipher is not used to encrypt numbers so it just leaves it as it is.