

Names: _____

Swift: Functions - Group Assignment

Part 1: Function Types

One of the most important parts of understanding functions is to first understand how to take your idea for a function, and represent it with the correct function type, i.e., the correct parameters and return types.

Below there are multiple English descriptions of a function. You should work together to determine the function declaration, including a good name for the function, the correct parameters types and names, and the correct return types.

The first one is done for you as an example.

Compute if a number is prime.

```
func isPrime(number: Int) -> Bool
```

Compute if a string ends in ".swift"

Compute the prime divisors of a number.

Compute if 2 numbers are relatively prime (they share no common divisors). Example: 9 and 8 are relatively prime, 9 and 6 are not relatively prime.

Compute if 3 floating point numbers are valid lengths for the sides of a triangle. Example: you can't make a triangle with lengths: 1.2, 1.3, 10.6.

Compute the mean, median, and mode of an array of floating point numbers

Sort an array of numbers

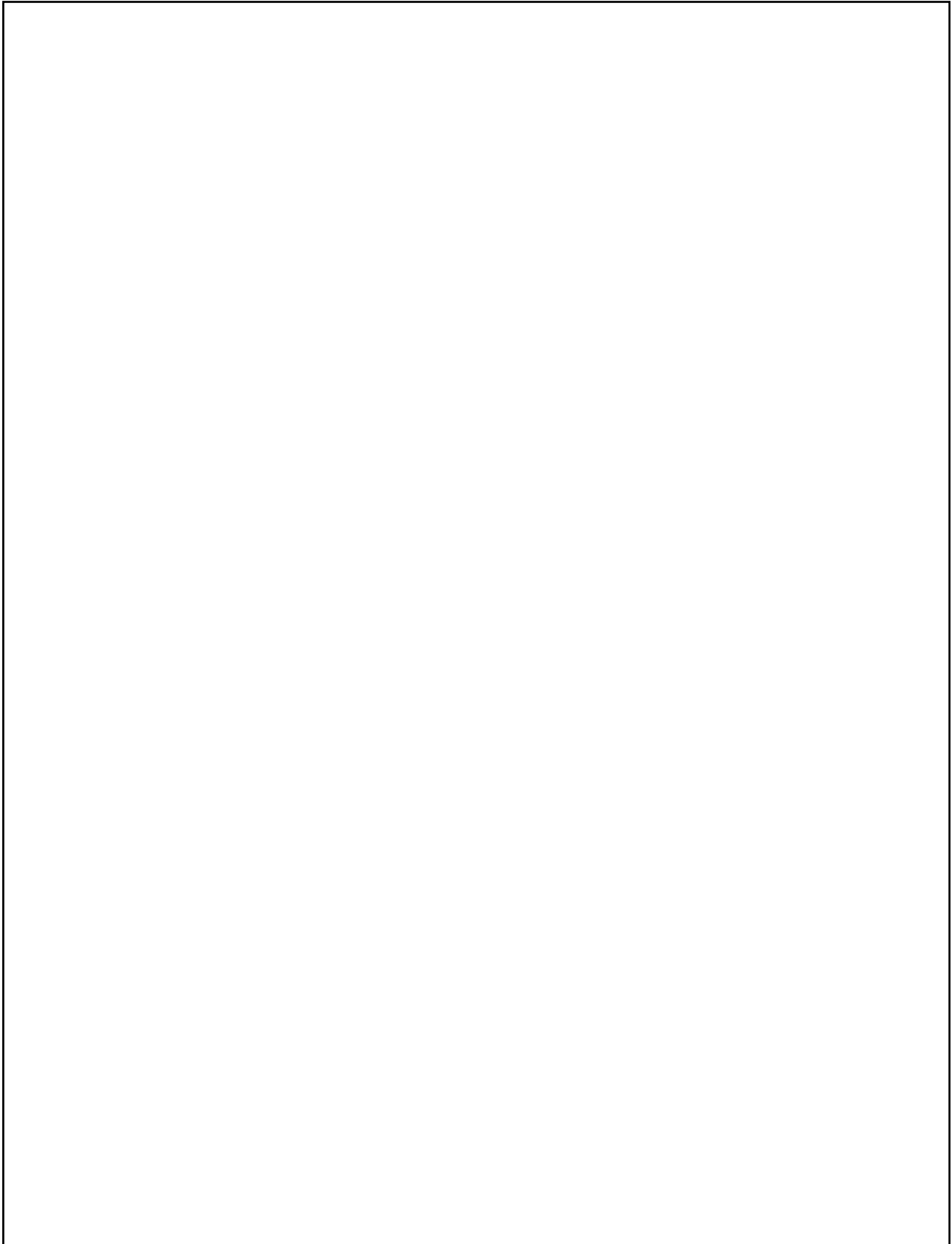
Part 2: Writing Functions

Now, work as a group to get practice writing complete functions.

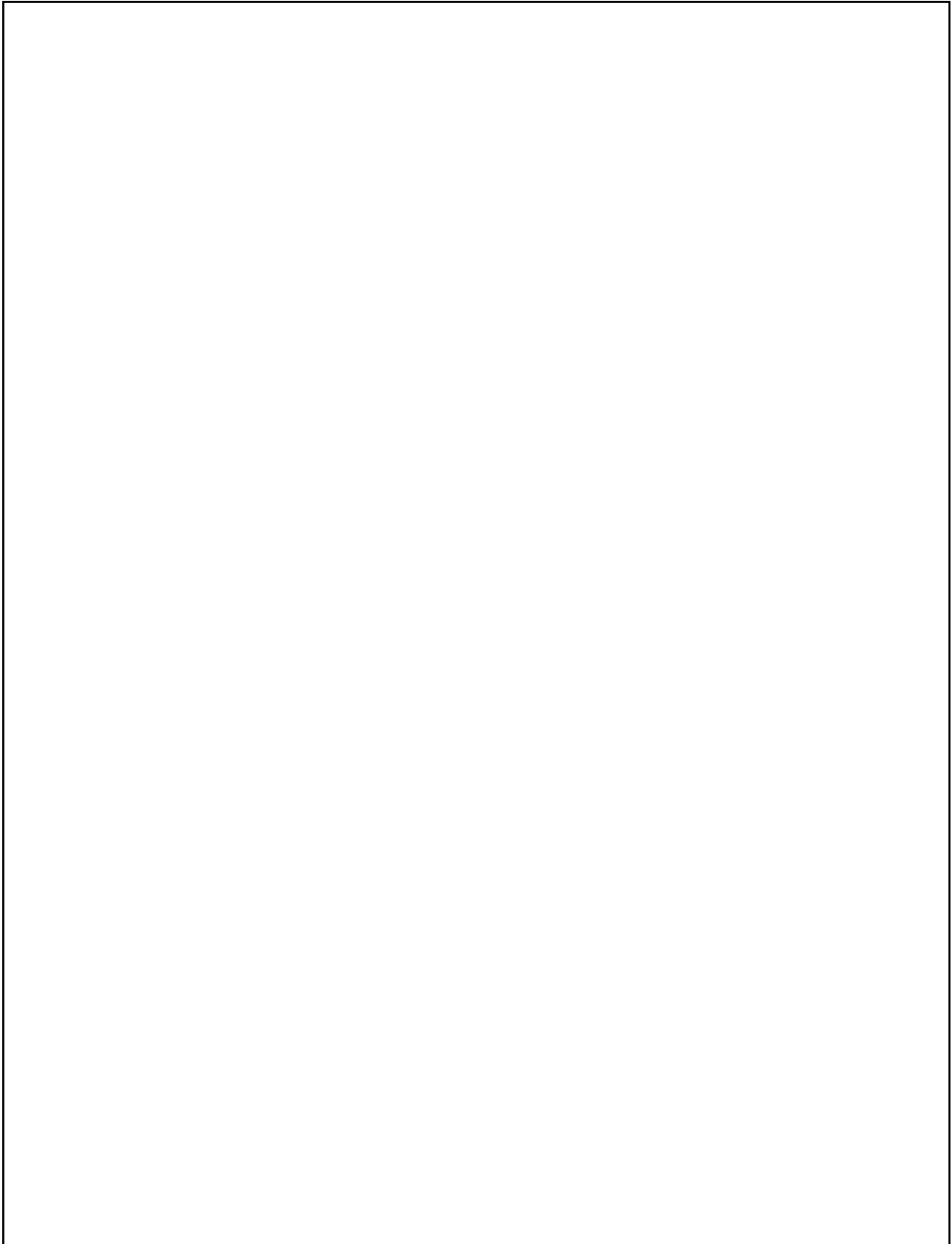
Pick 3 of the functions from above, and write the complete code for each. Some things are easy if you Google it.

Function 1

Function 2



Function 3

A large, empty rectangular box with a thin black border, occupying the majority of the page below the 'Function 3' header. It is intended for a drawing or detailed notes related to the function.

Part 3: Testing Functions

A very important aspect of writing functions is also testing that they work correctly. For **every** function in Part 1 (not just the ones you wrote actual code for), write some short code that tests each function with 5 different inputs to demonstrate that they work correctly.

The first one has been done for you as an example.

Compute if a number is prime.

```
print(isPrime(number: 1))
print(isPrime(number: 2))
print(isPrime(number: 3))
print(isPrime(number: 9))
print(isPrime(number: 269))
```

Compute if a string ends in ".swift"

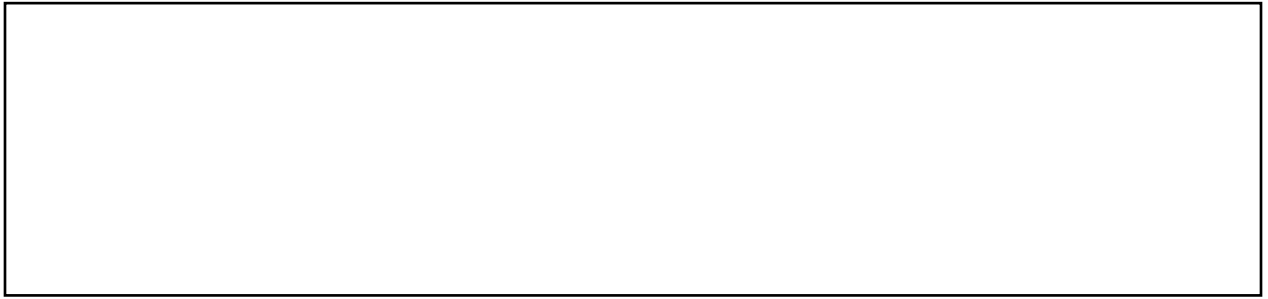
Compute the prime divisors of a number.

Compute if 2 numbers are relatively prime (they share no common divisors). Example: 9 and 8 are relatively prime, 9 and 6 are not relatively prime.

Compute if 3 floating point numbers are valid lengths for the sides of a triangle. Example: you can't make a triangle with lengths: 1.2, 1.3, 10.6.

Compute the mean, median, and mode of an array of floating point numbers

Sort an array of numbers

A large, empty rectangular box with a thin black border, intended for a student to write their solution or code for the sorting problem.