In Swift, functions can be passed as parameters to other functions and they can be returned from functions.

Question 1 options:

True

False

Which of the following expressions can be used to declare a function that takes an integer as a parameter and returns string?

Question 2 options:

func foo(a:Int)->String

function foo(a:Int)->String

func foo(a:Int):String

function foo(a:Int):String

Select the statement that is correct and the most complete describing Swift

Question 3 options:

Swift is a strongly typed language that allows for the type inference

Swift is a weakly typed language that allows for the type inference

Swift is a strongly typed language that does not allow for the type inference

Swift is a weakly typed language that does not allow for the type inference

In Swift, all types contain a nil as a possible value

Question 4 options:

True

False

Consider a variable x that is an integer. If you would like to print the value as part of a text e.g. for x=5, the printed text would be

Value of x is: 5

Which of the following expressions would you use?

Question 5 options:

print("Value of x is: \(x)")

print("Value of x is: "+String(x))

print("Value of x is: ", x)

All of the above

Which of the following is true?

Question 6 options:

In Swift, values are never converted (cast operation) automaticaly to another type

In Swift, values are always converted (cast operation) automaticaly to another type if needed

In Swift, convertion (cast operation) is not possible at all

There is no such thing as a cast operation in programming language

How the optional type is denoted in Swift?

Question 7 options:

by adding a question mark (?) after the name of the type

by adding an exclamation mark (!) after the name of the type

by adding a question mark (?) infront of the name of the type

by adding an exclamation mark (!) infront of the name of the type

Which of the following statement(s) **can** be used to declare an integer constant of value 5 in Swift?

Question 8 options:

let x = 5

let Int x = 5

const x:Int

x=5

const x = 5

Swift requires each line to be terminated by a semicolon ";"

Question 9 options:

True

False

Which of the following expressions allows us to declare an array?

Question 10 options:

var shoppingList = ["catfish", "water", "tulips", "blue paint"]

var shoppingList = {"catfish", "water", "tulips", "blue paint"}

array shoppingList = ["catfish", "water", "tulips", "blue paint"]

var shoppingList = array["catfish", "water", "tulips", "blue paint"]

How many times will the following loop execute?

```
for i in 1...10 {

  ...

}
```

Question 1 options:

5

9

10

11

Consider the following declaration and select the best (most complete) answer

```
func minMax(array:[Int])->(a:Int, b:Int)?
```

Question 2 options:

the function takes an array of Ints as a parameter and returns a tuple with two Int values or a nil value

the function takes an array of Ints as a parameter and returns a tuple with two Int values

the function returns a tuple with two Int values or a nil value

the function returns a Int value or a nil value

Swift allows for the functions to be nested (declare function in a body of another function)

Question 4 options:

True

False

Next Page

In the following declaration

```
func pow(firstArg a:Int)->Int
```

Question 5 options:

firstArg is the name used in the function call and **a** is used in the body

a is the name used in the function call and **firstArg** is used in the body

firstArg is the name used in the function call and **both firstArg** and **a** can be used in the body

both firstArg and **a** is the name used in the function call and **both firstArg** is used in the body

In Swift a class ...

Question 6 options:

can inherit from one class but conform to multiple protocols

can inherit from multiple classes and conform to multiple protocols can inherit from multiple classes but conform to only one protocol

can inherit from one class and conform to a single protocol

In the following declaration

```
func swapTwoInts(_ a: inout Int, _ b: inout Int) {
  (a, b) = (b, a)
}
```

Question 7 options:

_ means that when calling the function param labels should not be used

inout means that params are appearing and disappearing

_ means that params are output parameters

_ means that parameters are optional

Consider the following declaration and select the best (most complete) answer:

```
func add(no1:Int, no2:Int)->Int
```

Question 8 options:

it declares a function that takes two Int values as parameters and returns an Int value

it declares a function that takes two Int values as parameters

it declares a function that returns an Int value

this declaration is incorrect

Next Page

In Swift, function is identified by

Question 10 options:

keyword func

keyword function

returned type with ->

at least one parameter

SwiftUI can be only used to define UI for simple applications

Question 1 options:

True

False

The origin of the UI element in iOS app is the upper-left corner

Question 2 options:

True

False

Next Page

What is the name of the visual tool that is used to build layouts for the iOS application as well as to define navigation and transitions between screens in the application?

Question 3 options:

storyboard

SwiftUI

Layout editor

UI inspector

Next Page

SwiftUI uses a declarative approach

Question 4 options:

True

False

Next Page

What is SwfitUI?

Question 5 options:

It is an innovative way to build user interfaces using Swift

It is a UI component defined in Swift

It is a UI component allowing user to code in Swift

It is a scripting language that supports Storyboard

Next Page

●