1. All Swift classes must inherit from which root class

Not required

NSObject

NSRootObject

2. Choose an answer that is a way to unwarp optional:

Force Unwrapping (true)

Optional Binding (if let) (true)

Implicity Unwrapping (?)

All of the Above

3. Which of the following answer is not a properly way to declare a property

<pre><code class="language-swift"> var totalSteps: Int = 0 { willSet(newTotalSteps) { print(newTotalSteps) } } </code></pre> (true)

<pre><code class="language-swift"> let fourByFiveByTwo = Cuboid(width: 4.0, height: 5.0, depth: 2.0) </code></pre> (true)

<pre><code class="language-swift"> lazy let importer = DataImporter() </code></pre> (lazy var)

### <pre><code class="language-swift"> class var ComputedTypeProperty: Int { return 107 } </code></pre> (Type Property Syntax)

4. Which keyword will be used when a method want to change the value of a property in Struct or Enum?

indirect

typealias

internal

mutating

5. func swapNumber(a: inout Int, b: inout Int) {

a = a + b

b = a - b

a = a - b

}

var a = 1900

var b = 69

swapNumber(a: &a, b: &b)

print("a: \(a), b: \(b)")

What will be printed ?

a: 69, b: 1900

a: 1900, b: 69

a: 69, b: 1831

a: 1831, b: 69

6. Given a function that return a random value in the range from 1 to 5 How many time at least running this fuction to get number 5 with the accurate of 100% ?

5.0

25.0

100.0

None of the above

7. What type of Swift ?

Object Oriented Programming

Protocol Oriented Programming

Both Protocol Oriented Programming and Object Oriented Programming

Orther answers

8. The values in Set must be ?

Hashable

Int

String

Double

9. Given the following code:

var i = 2

repeat {

i \*= i \* 2

} while i < 20

print(i)

what will be the output ?

32.0

64.0

128.0

256.0

10. Given the following code:

let x = 1

let y = 0

let z = x != 2 || y == 4

print(z)

what will be the output ?

1.0

0.0

true

false

11. let array = [[true, false, true, true, false],

[false, true, true, false, true],

[true, false]

]

let result = array

.flatMap { $0 }

.reduce(true) { $0 && $1 }

print("result: \(result)")

What will be printed ?

result: false

result: true

runtime error

flatMap can be used in this case

12. What is type of Enum, Struct, Array ?

Value type

Reference type

Generic Type

Function type

13. Extension can not ?

Add stored type properties

Add computed type properties (true)

Adopt Protocol (true)

Define subscripts (true)

14. which way to prevent a method, property, or subscript from being overridden ?

Use the final keyword

Use the extention

Create private instance

Use the static keyword

15. Are structs thread-safe?

**Yes**

No

16. What is mean of encapsulation in OOP ?

Objects can hide the properties

Objects can hide the method

Project can be encapsulated

Object specifies whether or not to share its data to other objects

17. Given the following code:

let language: [String] = [“Swift”, ”Ruby”]

let pattern: [String] = ["Delegate"]

let sumary = [language ,pattern]

let flattenedSum = sumary.flatMap{ $0 }

let reducedSum = flattenedSum.reduce("", { $0 + $1 })

let finalSum = reducedSum.filter{ $0.count < 10}

print(finalSum)

What will be printed to the screen ?

Swift

SwiftRuby

Delegate

nil

18. Which is not different between Class and Struct ?

Inheritance enables one class to inherit the characteristics of another.

Deinitializers enable an instance of a struct to free up any resources it has assigned.

Type casting enables you to check and interpret the type of a class instance at runtime.

Reference counting allows more than one reference to a class instance.

19. Which of the following answers contain only Value Type ?

Class, Closure, Tuple

Struct ,Enum, Closure

Enum, Tuple, Struct, Protocol

Tuple, Closure, Enum, Protocol

20. Given:

//-1 + 0 = -1

1+2 = 2

3+4 = 9

5+6 = 20

9+10 = ?

Choose an answer below

120.0

69.0

54.0

39.0

1, Which of the following statments is true about the similar and different between Struct and Class ?

Conform to protocols to provide standard functionality of a certain kind (sai : same)

Define subscripts to provide functionality to their values using subscript syntax ( sai : same)

Type casting enables you to check and interpret the type of a class instance at anytime. (anytime)

Deinitializers enable an instance of a class and struct to free up any resources it has assigned. (sai)

2. In the following reasons, which is the reason for memory leak?

Too much data are saved in cache

Retain circle

Using static variable

Too much global variables are used

3. Select right answer about array, set and dictionary.

An array stores values of the same type in an ordered list. One value can only appear once (false)

A set stores non-distinct values of the same type in a collection with no defined ordering. (false : A set stores distinct values of the same type in a collection with no defined ordering.)

A type must be hashable in order to be stored in a set (true)

"Key" and "Value" must be hashable type in order to be stored in a dictionary (false : Key must be Hashable types, Value can be any kind of types)

4, What is quality of services Concurrent queue are ordered from hight to low ?

UserInitiated, UserInteractive, Utility, Background

UserInteractive, UserInitiated, Utility, Background

UserInitiated, UserInteractive,Background, Utility

UserInteractive, UserInitiated, Background, Utility

5, Does Swift have multiple inheritance?

Yes

No

6, What is the mechanism of delegate pattern ?

Hear - call

Authorization

Synchronized

Asynchronous

7, What is the effect of the weak keyword?

Increase retain count to 1

Decrease retain count to 1

Does not increase retain count

Increasing or decreasing retain count depends on the available memory

(https://learnappmaking.com/weak-vs-strong-references-swift/)

8. var a = 15

var b = 0

a < 10 ? b = 15 : a > 5 ? (b = -1) : (b = 7)

Which is value of b?

-1

7

15

0

9, What is Optional chaining ?

It is a process for querying and calling properties, methods, and subscripts on an optional that might currently not be nil

Change optional from nil to a specific value

Change the value of optional to nil

It is a process for querying and calling properties, methods, and subscripts on an optional that might currently be nil

10, Which of the following stament is true about Overload's definition when fill in the blank ? "the process of creating two or more than two functions with the same name but having ..."

different number or types of parameters returned

different number or types of parameters passed

different number or types of parameters passed but same number or types of parameters returned

same number or types of parameters passed but different number or types of parameters returned

11, Which of these following symbols does not refer to Optional ?

! (true)

? (true)

!!

?? (true)

12, Given the following code:

var x = 0

repeat {

print(x++)

print(++x)

} while x < 9

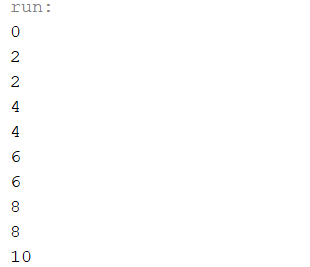
what is the 5th element to be printed to the screen ?

2.0

4.0

6.0

8.0



13, Given the following function: func sort(\_ array: [Int]) -> [Int] { guard array.count > 1 else { return array } let middleIndex = array.count / 2 let leftArray = sort(Array(array[0..

Merge Sort

Heap Sort (https://dnmtechs.com/thuat-toan-sap-xep-vun-dong-heap-sort/)

Quick Sort

Bucket Sort