Marks - 3

1. **Correct the sorting function such that it must first sort by 'type' and then by 'color'.**

For example, the above code should output - redApple, greenApple, orangeOrange, yellowBanana

```
struct Fruit {
   let type: FruitType
   let color: FruitColor
}

enum FruitType: Int {
   case Apple = 40
   case Orange
   case Banana
}

enum FruitColor: Int {
   case red = 41
   case green
   case yellow
   case orange
}

var fruitArray = [Fruit]()
let yellowBanana = Fruit(type: .Banana, color: .yellow)
let redApple = Fruit(type: .Apple, color: .red)
let orangeOrange = Fruit(type: .Orange, color: .orange)
let greenApple = Fruit(type: .Apple, color: .green)

fruitArray.append(yellowBanana)
fruitArray.append(orangeOrange)
fruitArray.append(greenApple)
fruitArray.append(redApple)

fruitArray.sort(by: {
   /// Correct the logic here
   return $0.color.rawValue < $1.color.rawValue
})
```

```
if $0.color == $1.color {
    return $0.type.rawValue > $1.type.rawValue
} else {
    return $0.color.rawValue > $1.color.rawValue
}
```

```
if $0.type == $1.type {
  return $0.color.rawValue < $1.color.rawValue
} else {
  return $0.type.rawValue < $1.type.rawValue
}
```

```
if $0.type == $1.type {
    return $0.color.rawValue > $1.color.rawValue
} else {
    return $0.type.rawValue > $1.type.rawValue
}
```

```
if $0.color == $1.color {
    return $0.type.rawValue < $1.type.rawValue
} else {
    return $0.color.rawValue < $1.color.rawValue
}
```

Marks - 2

2. Does the code snippet below have any errors ?

```
func greetCandidate(_ name: String = "Anonymous") {
        // Do Something
}
let greetCopy = greetCandidate
greetCopy("Candidate")
```

   - Yes
   - **No**

Marks - 1

3. What are two properties that Auto Layout constraints control on a UIView? Choose as many as you like

   - **Size**
   - Alpha
   - Depth
   - **Position**

- Rotation

Marks - 1

4. How do you center a 100x100 point profile image in the top third of an app?
   Choose as many as you like

   - Pin a image to the center of the main screen, and adjust the constraints to move it up -250 points
   - **Pin a view to the top third, and add an image as a subview that is centered within this top view**

Marks - 1

5. What does it mean for a UILabel to be relative to a UIImageView?
   Choose as many as you like

   - **The UILabel's position will be dependent position and size of the UIImageView**
   - The UILabel will move with the UIImageView as it animates
   - The content fill type of the UIImageView will change the position of the UILabel

Marks - 1.5

6. What are size classes?
   Choose as many as you like

   - **Rough approximation of screen type**
   - **Provide layout control in different orientations**
   - **Enable UI customizations based on type of device**
   - Fast rendering graphical views

Marks - 2

7. How do points relate to pixels?
   Choose as many as you like

   - **On an iPhone 8 Plus, 1 point equals 3 pixels**
   - Points are an abstraction of pixels
   - On an iPhone 7, 1 point equals 1 pixel
   - **On an iPhone 8, 1 point equals 2 pixels**

Marks - 2

8. class GoScale {
   var name: String?

```
        struct GoScaleStruct {}
}
let classInstance = GoScale()
let structInstance = GoScale.GoScaleStruct()
```

Which of the following holds true for the above code snippet

- classInstance is stored in Heap, structInstance is stored in Stack
- classInstance is stored in Stack, structInstance is stored in Heap
- **classInstance is stored in Heap, structInstance is stored in Heap**
- classInstance is stored in Stack, structInstance is stored in Stack

Marks - 1.5

9. Strong references increase the retain count of an object by 1, weak references...

    - increase the retain count of an object by 1
    - decrease the retain count of an object by 1
    - **zero out the pointer to your object when it successfully deallocates**

Marks - 1.5

10. What output will be produced by the code below?

```
struct TaylorFan {
        static var favoriteSong = "Shake it Off"
        var name: String
        var age: Int
}
let fan = TaylorFan(name: "James", age: 25)
print(fan.favoriteSong)
```

    - "Shake it Off"
    - nil
    - Nothing will be output
    - This code will compile but crash
    - **This code will not compile**

Marks - 1.5

11. When this code is executed, what will result be set to?

```
func fetchGoScaler() -> (job: String, name: String) {
        return (“iOS Developer”, “Meh”)
}
let result = fetchGoScaler().0
```

    - “Meh”

- **"iOS Developer"**
- ("iOS Developer", “Meh”) (a tuple)
- (0) (a tuple)
- ["iOS Developer", “Meh”] (an array)
- This code will compile but crash
- This code will not compile

Marks - 2

12. To declare a static property or function you use the static modifier on value types. Which of the following statements are true for reference types(classes) ?

    - When applied to classes, static becomes an alias for class.
    - When applied to classes, static becomes an alias for final.
    - **When applied to classes, static becomes an alias for class final.**

Marks - 1

13. The **Frame** of an UIView is the rectangle, expressed as a location (x,y) and size (width,height) relative to its own coordinate system (0,0).
    The **Bounds** of an UIView is the rectangle, expressed as a location (x,y) and size (width,height) relative to the superview it is contained within

    - **Above Statement is incorrect**
    - Above Statement is correct

Marks - 1.5

14. ***layoutIfNeeded*** *says* **update immediately please***,*
    whereas ***setNeedsLayout*** *says* **please update but you can wait until the next update cycle**.

    - Above Statement is incorrect
    - **Above Statement is correct**

Marks - 1.5

15. **Any** is for value types
    Any is for reference types

    - Both Statements are incorrect
    - Statement 1 is correct
    - **Both Statements are correct**
    - Statement 2 is correct

Marks - 1.5

16. Int is a value type
    NSString is a reference type

Closures are of value types

- All the Statements are correct
- **Statement 1 and 2 are correct**
- Statement 1 and 3 are correct
- Statement 2 and 3 are correct

Marks - 1

17. In a horizontal UIStackview having 2 UILabels, setting Content Hugging Priority of first UILabel to 1000 means the text in that UILabel will not truncate.
    - **Above Statement is incorrect**
    - Above Statement is correct

Marks - 1

18. Which is correct for Enumerations?

    - **Enumerations can define initializers.**
    - Enumerations cannot define initializers.
    - Enumerations cannot conform to protocols.

Marks - 1

19. What is true for lazy variables?

    - **The memory gets allocated once you use/access the lazy variables**
    - The memory gets allocated before you use/access the lazy variables
    - **It is var and not let**

Marks - 1

20. Does struct support inheritance in swift?

    - **No**
    - Yes