

What is the output of the following statement?

```
>>> val a = '7'
>>> val b = 7
>>> val c = "7"
>>> println("$a, $b, $c")
```

- A '7', 7, "7"
- B 7, 7, 7
- C 7, 7, "7"
- D something else

How often does the following code print "CS109 is fun"?

```
>>> fun element(): Int {
...   println("CS109 is fun")
...   return 13
... }
>>> val a = Array(7) { element() }
```

- A never
- B once
- C seven times
- D 13 times

Where are the errors in the following code?

```
>>> data class Point(var x: Int, var y: Int)
>>> val a = Point(2, 3)
>>> var b = Point(3, 4)
>>> val c = a
>>> var d = b
>>> a.x = 7           // A
>>> println(c)
>>> b.x = 9           // B
>>> d = a             // C
>>> println(d)
>>> c = b             // D
>>> println(c)
```

- A Error in A, B, C, D
- B Error in A
- C Error in A, D
- D Error in A, C, D
- E Error in C, D
- F Error in D

True or False?

Each of the following code pieces creates an object on the heap.

```
>>> 27
>>> "CS109 is fun"
>>> { x: Int -> 3 * x - 27 }
>>> 13.5
```

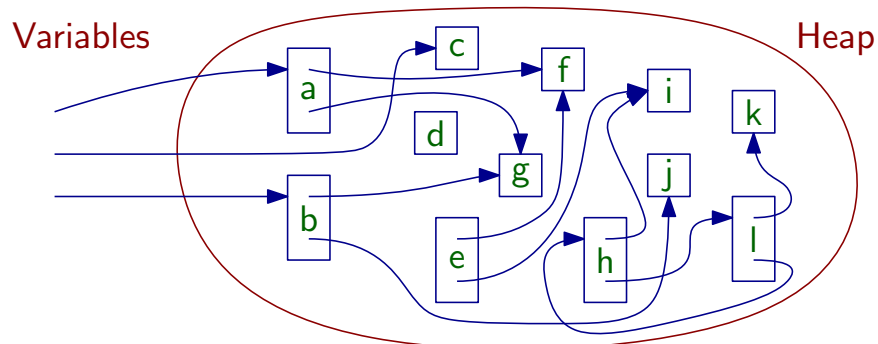
- A True
- B False
- C Don't know

What is the output of the following statements?

```
>>> val a = setOf(2, 3, 9, 13)
>>> val b = mutableSetOf<Int>()
>>> b.add(3)
>>> b.add(13)
>>> println(a - 2 == b + 9)
>>> println(a.containsAll(b))
>>> println(2 in b)
>>> println(a - b)
```

- A false, false, false, [2, 9]
- B true, false, false, [2, 9]
- C false, true, false, [2, 9]
- D true, true, false, [2, 9]
- E true, true, true, [2, 3, 9, 13]
- F false, true, true, [2, 3, 9, 13]
- G false, true, false, [2, 3, 9, 13]

Which objects are garbage?



- A d
- B d, e
- C d, e, f
- C d, e, f, i
- D d, e, h, i, j, k, l
- E d, e, h, i, k, l
- F Don't know

What is the output of the following statements?

```
fun test(s: String?) {
    println(s ?: "X")
    println(s?.length ?: -1)
    println(s!!)
}
test("CS109")
test(null)
```

- A CS109 5 CS109 null 0 null
- B CS109 5 CS109 null -1 null
- C CS109 5 CS109 X -1 null
- D CS109 5 CS109 X -1 error message
- E Don't know

What is the output of the following statement?

```
val m = mapOf(2 to "two", 3 to "three")
println(m[3].length)
```

- A 5
- B 3
- C 13
- D error in compilation
- E runtime error
- F Don't know

The `getOrElse` method is used like this:

```
>>> val m = mapOf(3 to "three", 7 to "seven")
>>> m.getOrElse(3) { "iii" }
three
>>> m.getOrElse(4) { "iv" }
iv
```

What are the types of the parameters in `m`'s `getOrElse` method?

- A `Int`
- B `Int, String`
- C `Int, (Int) -> String`
- D `Int, () -> String`
- F Don't know

Everyone knows that debugging is twice as hard as writing a program in the first place. So if you're as clever as you can be when you write it, how will you ever debug it?

Brian Kernighan

Debugging is like being the detective in a crime movie where you are also the murderer.

Filipe Fortes