

1 (e) The `main` method should have three variables. The `Fraction` class should be shown with the `isQuoted` attribute. There should be two `Fraction` objects in memory.

		main
f	⇒	400
g	⇒	500
h	⇒	500
		Fraction class
isQuoted	⇒	false
	400	Fraction instance
numerator	⇒	1
denominator	⇒	2
	500	Fraction instance
numerator	⇒	3
denominator	⇒	4

2. parts (e) and (f) illustrate the short-circuit evaluation that Java performs for logical expressions.

2 (d)

```
x < y && z < y
(10 < 15) && z < y
true && z < y
true && (0 < 15)
true && true
true
```

2 (d)

```
x > y && z < y
(10 > 15) && z < y
false && z < y      ( z < y is not evaluated )
false
```

2 (e)

```
x < y || z < y
(10 < 15) || z < y
true || z < y      ( z < y is not evaluated )
true
```

5 (a) The state of an object is defined by the values of its attributes. The identity of an object is its location in memory. The behavior of an object is defined by its methods.

5 (d) A public attribute becomes a permanent part of the API of a class because clients will write code that uses the attribute; if the class implementer changes the attribute (makes it private, changes its type, changes its value, etc) all of the client code that uses the attribute will potentially break. This is a bad thing. If instead the attribute were private, and a public accessor was provided to return the value of the attribute, then the class implementer can change the hidden representation of the attribute without breaking client code.

6.

- `Fraction` is used in `main` but not imported
- `static` is missing from `main` method header
- semicolon missing after `System.out`
- `scanner` should be `Scanner`
- a loop is used to compute `f` raised to the power `exp`; there is a method called `pow` in the `Fraction` class that we can delegate to instead
- the loop is incorrect; it actually computes `f` raised to the power 2^{exp}
- `f > max` and `f < max` will not compile because you cannot use operators (`>` and `<`) with object references; the client should have used the `compareTo` method instead
- `output.printf` cannot print `Fraction` objects with the `%f` conversion; use `%s` instead (which will print the fraction as a string)