

EECS 1022 3.0 Programming for Mobile Computing

Solution of Midterm - Version D

18:30–19:30 on July 10, 2017

1 (2 marks)

- (a) The process of software development consists of multiple phases. What is the responsibility of the tester?

Answer: Checking whether the code satisfies the specification.

Marking scheme: 1 mark for mentioning checking that the *code* satisfies the *specification* or is *correct*.

- (b) Which phase is before the testing phase?

Answer: Implementation.

Marking scheme: 1 mark for implementation.

2 (2 marks)

The design pattern MVC separates the code into a model, a view, and a controller. Assume you develop a mobile app in which the user is shown a sentence and has to type it in word-for-word and determines the user's typing accuracy and speed. For each of the following items, indicate whether it belongs to the model, the view or the controller.

- (a) The code that computes the typing speed.

Answer: model.

Marking scheme: 0.5 mark for model.

- (b) The text box in which the user types the sentence.

Answer: view.

Marking scheme: 0.5 mark for view.

- (c) The sentence displayed to the user.

Answer: model.

Marking scheme: 0.5 mark for model.

- (d) The code that extracts the typed sentence from the text box.

Answer: controller or activity.

Marking scheme: 0.5 mark for controller or activity.

3 (2 marks)

Consider the API of the class `Oven` which is provided at the end of this test.

- (a) How many (public) constructors does the class have?

Answer: 3.

Marking scheme: 1 mark for 3.

- (b) What is the return type of the `flipSwitch` method?

Answer: void or none.

Marking scheme: 1 mark for void or none.

4 (2 marks)

Can two objects have the same state but different identities? **Explain your answer.** You only get marks for your explanation.

Answer: Yes. For example, consider

```
Person one = new Person("First", "Last", 1);
Person another = new Person("First", "Last", 1);
```

The objects created above, referenced by `one` and `another`, have that same state since the attributes have the same values, but have different identities as the objects reside at different memory addresses.

Marking scheme: 1 mark for mentioning that there can be two objects whose *attributes* have the same *values*. 1 mark for mentioning that these objects reside on different *memory addresses*.

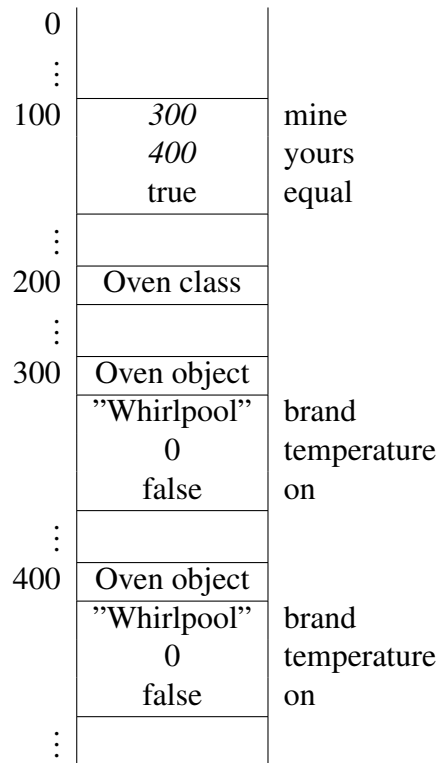
5 (2 marks)

- (a) Consider the following code snippet.

```
Oven mine = new Oven("Whirlpool", 0)
Oven yours = new Oven("Whirlpool");
boolean equal = mine.equals(yours);
```

Draw the corresponding memory diagram. Make sure that the attributes `brand`, `temperature` and `on` of the `Oven` class and the variables `mine`, `yours` and `equal` are reflected in your diagram.

Answer:



"Whirlpool" is a String object and, hence, should be represented by an object block as well, but for simplicity we have not done that here.

Marking scheme:

- 0.25 mark for two Oven object blocks.
- 0.25 mark for the values of `mine` and `yours` are the addresses of the two objects.
- 0.25 mark for the value of `equal` variable.
- 0.25 mark for the correct values of the attributes of the Oven objects.

(b) The method `flipSwitch` is implemented as follows.

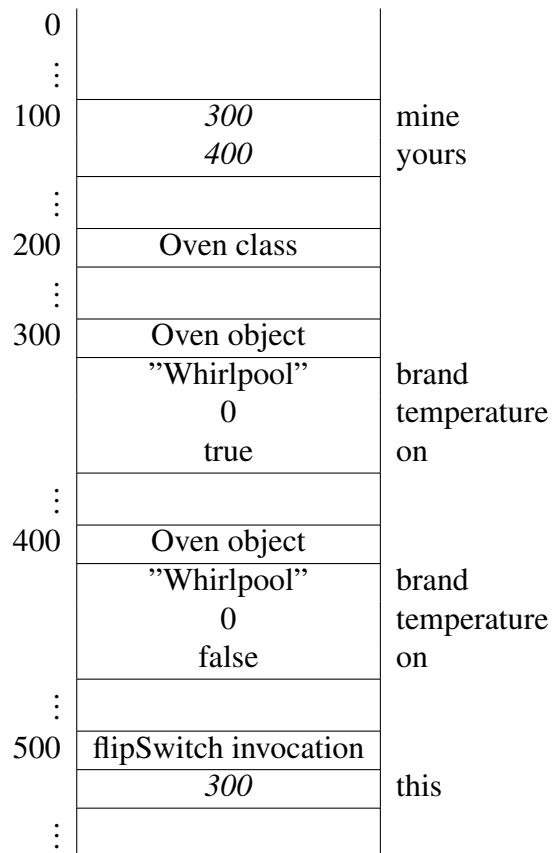
```
public void flipSwitch()
{
    this.on = !this.on;
}
```

Consider the following code snippet.

```
Oven mine = new Oven("Whirlpool", 0)
Oven yours = new Oven("Whirlpool");
mine.flipSwitch();
```

Draw the corresponding memory diagram. Make sure that the attributes `brand`, `temperature` and `on` and the variables `mine` and `yours` are reflected in your diagram. Make sure to include the invocation block for the call of the `flipSwitch` method.

Answer:



"Whirlpool" is a String object and, hence, should be represented by an object block as well, but for simplicity we have not done that here.

Marking scheme:

- 0.33 mark for this in the invocation block.
- 0.33 mark for the value of this (address of the Oven object referred to by mine) in the invocation block
- 0.33 mark for the new value (true) of the on attribute of the Oven object referred to by mine.