

CS100J, Fall 2006 Preparing for Prelim 2:

Tuesday, 17 October, 7:30—9:00PM, Olin 155
Review session, Sunday, 15 October, 1–3PM, Philips 101

This handout explains what you have to know for the second prelim.

Material from the first prelim. The second prelim will cover all the material that was on the first prelim. This material was discussed on the handout for the first prelim. Please look at that handout (you can get it from the course website if you lost it).

In addition, you need to know the following.

1. Constructors. Secs. 3.1.3 and 4.1.3 of the text

- Know what a constructor is.
- Be able to write down how to evaluate a new expression —and also be able to do evaluate one yourself.
- Be able to write a constructor that calls another constructor in the same class (using **this**) or calls a constructor in a superclass (using **super**).

2. Apparent and real classes. Casting from a subclass to a superclass and back. See Sec. 4.2 of the text.

3. The class hierarchy. Class Object. Function equals. Operator instanceof. See Sec. 4.3.

4. Methods — executing method calls, pages 93-94. You need not know the later material on the stack of frames for a method call.

- Be able to draw a frame for a method call, knowing what goes in the scope box depending on the method being called.
- Be able to write down the steps involved in executing a method call (see below)
- Be able to execute a method call by hand, drawing any frames and folders as necessary.

Here are the steps involved in executing a method call:

1. Draw the frame —including everything that goes in it (see page 93).
2. Assign the values of the arguments to the parameters (in the frame).
3. Execute the method body.
4. Erase the frame. If this is a function, yield the value of the expression in the return statement as the value of the function call.

5. Recursion

Be able to write recursive functions.

6. Loops

- Know the syntax and semantics of the for-loop.
- Be able to execute a for-loop that we give you.
- Be able to solve simple problems that require a for-loop that processes a range of integers.
- Be able to write an invariant for a problem that requires a for-loop that processes a range of integers.