

Page: 1 Name: _____

CS56—Midterm Exam
E01, W16, Phill Conrad, UC Santa Barbara
Wednesday, 02/10/2015

Name: _____

Umail Address: _____@ u mail. ucsb. edu

- Please write your name **above AND AT THE TOP OF EVERY PAGE**
 - Please put your pages **in order, facing the same way.**
 - All the odd pages have dots (•); these should be upper right, and facing up.
 - All the even numbered pages have crosses (×) at upper right and should be facing down.
 - Be sure you turn in every page of this exam.
 - Each of the pages is numbered (e.g. Page 1, Page 2, etc.)
 - The last page clearly says "End of Exam".
 - This exam is **closed book, closed notes, closed mouth, cell phone off**
 - You are permitted **one sheet of paper** (max size 8.5x11") on which to write notes
 - This sheet will be collected with the exam, and might not be returned
 - Please write your name on your notes sheet
-



1. (50 pts) In the space below (and on the next page if needed) write the code for the `IndexEntry` class according to the instructions on the separate [handout](#) provided.

There is more room on the next page for your answer if you run out of room here.



Page: 3 Name: _____

Extra space for your answer to question 1



2. (10 pts) For each of the following indicate if the line of code involves auto-boxing, and/or auto-unboxing. If a line of code involves both, check both boxes. If it involves neither, check neither box. ASSUME THAT ALL THE LINES OF CODE ARE IN THE SAME main METHOD, CONSECUTIVELY.

Code	auto- boxing	auto- unboxing
<code>ArrayList<Integer> mylist = new ArrayList<Integer>();</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>mylist.add(new Integer(7));</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>mylist.add(2);</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>mylist.add(mylist.get(0));</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>Integer x = mylist.get(0);</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>int y = mylist.get(1);</code>	<input type="checkbox"/>	<input type="checkbox"/>
<code>Integer z = mylist.get(mylist.get(1));</code>	<input type="checkbox"/>	<input type="checkbox"/>

Grading: -2 for each incorrect answer, but no more than -10 total.)

3. (6 pts) (Continued from previous problem). Assume that all of the lines of code in the previous problem appear in a main method, followed by these lines of code. Indicate what the output will be.

(Note: If you believe the code will throw an exception when run, you do not need to replicate the *exact* format of the error message as it would appear in the output; an approximation of it is good enough.)

```
System.out.println("x=" + x);
System.out.println("y=" + y);
System.out.println("z=" + z);
```

4. (4 pts) Consider the statement: "Java has automatic garbage collection."

BRIEFLY, but precisely describe what this statement means.

**For all the questions on this page:**

- Answer in the context of Java programming
- The questions are all multiple choice, and provide you with several statements about a variable, ONLY ONE of which is COMPLETELY TRUE. (Some others may be partially true, and partially false.) Choose the one COMPLETELY true answer.

5. (4 pts) Suppose that you have a variable `sum` of type `int` declared inside the `main` method of a class. Which statement is *completely* true about this variable?

a. It is a reference variable and it is also a local variable.

So the reference is stored on the stack, but the object referred to is on the heap.

b. It is a primitive variable, and it is a local variable. So it is stored on the heap.

c. It is a primitive variable, and it is a local variable. So it is stored on the stack.

d. It is a primitive variable, and it is also an instance variable. So the value is stored on the heap.

e. It is a reference variable, and it is also an instance variable. So the reference is stored on the stack, but the object referred to is stored on the heap.

6. (4 pts) This question refers to the `IndexEntry` class that you wrote for question 1, which has a private data member called `topic` of type `String`.

Suppose you instantiate that class in a `main` with this line of code:

```
IndexEntry ie = new IndexEntry("byte", 51);
```

Which statement is *completely* true about the private data member `topic` of the object `ie`?

a. It is a reference variable and it is also a local variable.

So the reference is stored on the stack, but the object referred to is on the heap.

b. It is a primitive variable, and it is a local variable. So it is stored on the heap.

c. It is a reference variable, and it is also an instance variable.

So the reference itself is on the heap, and so is the object to which the reference refers.

d. It is a primitive variable, and it is a local variable. So it is stored on the stack.

e. It is a primitive variable, and it is also an instance variable. So the value is stored on the heap.

f. It is a reference variable, and it is also an instance variable.

So the reference is stored on the stack, but the object referred to is stored on the heap.

7. (4 pts) Continuing from the previous question—now consider the variable `ie` in the line of code that instantiates `IndexEntry`—the line that we said appears in the `main` method. Which statement is *completely* true about `ie`?

a. It is a reference variable and it is also a local variable.

So the reference is stored on the stack, but the object referred to is on the heap.

b. It is a primitive variable, and it is a local variable. So it is stored on the heap.

c. It is a primitive variable, and it is a local variable. So it is stored on the stack.

d. It is a primitive variable, and it is also an instance variable. So the value is stored on the heap.

e. It is a reference variable, and it is also an instance variable.

So the reference is stored on the stack, but the object referred to is stored on the heap.

8. (4 pts) Chris Gaucho wrote this code:

```
Student s = new Student("Phill",1234567);  
System.out.println("s=" + s);
```

Chris ran the code and got this output:

```
s=Student@e13e07
```

What Chris wanted to see was this:

```
s=(name: Phill, perm: 1234567)
```

What part of Chris' class is missing? You don't need to write the full Java code—just explain in English (but using precise Java terminology) what Chris needs to do to fix this problem.

9. The three steps that are typically involved in getting code into a git repo are these. Note that I've put ... where something normally goes.

```
git add ...  
git commit ...  
git push origin master
```

a. (4 pts) What typically follows `git add` on the command line? Briefly describe.

(Your description may include an example, but an example alone is NOT ENOUGH for full credit. I want a general description.)

b. (4 pts) What typically follows `git commit` on the command line? Briefly describe.

(Your description may include an example, but an example alone is NOT ENOUGH for full credit. I want a general description.)

c. (4 pts) What does the `git push origin master` step do that is NOT accomplished by only doing the two previous steps?

10. In C/C++ programming, we used the `make` utility to streamline the process of compiling code.

a. (1 pts) What is the name of the utility we are using in this course instead of `make`, because it is more suitable for Java?

b. (1 pts) With `make` we use a file called `Makefile` to control our build. What's the corresponding file called with the utility we are using in this course?

End of Exam

total points=100