

First name (color-in initial)	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	section (2,3,or 9)	first name initial	last name initial
Last name (color-in initial)	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z			

# H14: Due Tue 02.21 in Lecture Total Points: 50

MAY ONLY BE TURNED IN DURING THE CLASS INDICATED ABOVE, or offered in person, for in person grading, during instructor or TAs office hours. See the course syllabus at <https://foo.cs.ucsb.edu/56wiki/index.php/W12:Syllabus> for more details.

(1) (10 pts) Fill in the information below. Also, fill in the A-Z header in the usual way\*.

- i.e. **coloring in** the first letter of your first and last name (as it would appear in Gauchospace), writing **either 2, 3 or 9** to indicate your **discussion section** meeting time, and writing your **first and last initial** in large capital letters. All of this helps us to manage the avalanche of paper that results from the daily homework.

name:	
umail address:	@umail.ucsb.edu

## Reading Assignment:

In addition to HFJ (Head First Java). The other textbook for the course is the Java Pocket Guide, which I'll refer to as **JPG**. Both HFJ and JPG have their own pages on the wiki with reading notes.

- Read JPG :
  - Chapter 1, (pp. 3-5) (On Campus Off Campus)
  - Chapter 2, pp (6-16) (On Campus Off Campus)
  - Chapter 3, pp (17-25) (On Campus Off Campus)

Also, briefly skim the wikipedia article: [http://en.wikipedia.org/wiki/IEEE\\_754-1985](http://en.wikipedia.org/wiki/IEEE_754-1985) so that when Chapter 3 mentions the IEEE\_754-1985 floating point standard, you'll know what they are talking about.

- A side note: there is a new version of IEEE\_754-1985 called IEEE\_754-2008—but we'll put off discussing that to another day (i.e. a later course).
- This is the last homework before Midterm E02.** We will grade it and give it back to you in your assigned lab this week (before the exam.)

(Please turn to page 2)

A note about the questions on this assignment.

The Java Pocket Guide (JPG) is a reference guide, not a textbook. Therefore:

- Some of the questions I'm asking below "big picture" things about Java—things you should definitely know for the exam, and more importantly, for your career in Software Development or Computer Science.
- Other questions are nitpicky details that might be things you might not be expected to memorize—I've included them to give you practice in using a language reference.

Deciding which is which is a judgement call, and learning to make that judgement call is part of what I'm giving you the opportunity to learn in this course. I'm not going to play the "is this going to be on the exam" game, where I'm asked to commit to inclusion/exclusion of every fact or topic—Instead, I'm going to trust that you can put your critical thinking skills to good use, and exercise good judgement in figuring out what is an important theme vs. what is a nitpicky detail.

(2) Java Pocket Guide (Ch 1) mentions some naming conventions that are common practice among Java Developers. **According to the author:**

- (2 pts) What part of speech (noun, verb, adjective, adverb, etc.) should you use for class names?
- (2 pts) What part of speech should you use for interface names?
- (2 pts) If an interface name describes some capability, what letters should the name end with?
- (2 pts) What part of speech should you use for method names?
- (2 pts) What is an exception to that rule?

(Please turn to page 3)

(3) From JPG Ch2, concerning literals in Java:

- (2 pts) Briefly: in general, what is a "literal" in a programming language?
  
- (2 pts) How does Java distinguish between character and string literals?
  
- (2 pts) How do we express integer literals in hex? Give an example.
  
- (2 pts) How do we express integer literals in octal? Give an example.
  
- (2 pts) What is the name of the character set used by Java?

(4) (3 pts) Briefly explain: What do we mean by a Java **identifier**?

(5) (2 pts) In addition to upper and lower case letters, the underscore and digits (after the first character), what other character is legal in a Java identifier?

(6) (2 pts) What is the meaning of the ^ operator in Java?

(Please turn to page 4 for more)

(7) (3 pts) The discussion next to Table 2-6 makes the point that you should use `println()` rather than an escape sequence (e.g. `\n`) for making line breaks. What is the big picture thing that is going on here—i.e. what can you learn from this discussion that might be relevant not only to programming in Java, but dealing with line breaks in any programming language?

From JPG Chapter 3:

(8) There are several special floating point values that Java provides to be in compliance with IEEE\_754-1985. Skim over the Wikipedia article: [http://en.wikipedia.org/wiki/IEEE\\_754-1985](http://en.wikipedia.org/wiki/IEEE_754-1985) as well as section 3.3, and briefly describe the purpose of:

- (2 pts) `Double.NEGATIVE_INFINITY`

- (2 pts) `Double.NaN`

(8) (2 pts) What is meant by "number promotion" in a programming language?

(9) (2 pts) Write a line of Java code that will perform auto-unboxing.

(10) (2 pts) Write a line of Java code that will perform auto-boxing.

(End of H14)