

CS56-W15-H17 page 1

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 (4, 5 or 6)	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			

H17: Due Wednesday, 02.11 in Lecture

More on Data Structures (HFJ Ch 16)

Assigned: Thu 02.05

Total Points: 50

MAY ONLY BE TURNED IN IN THE LECTURE/LAB LISTED ABOVE AS THE DUE DATE, 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/W15:Syllabus> for more details.

(1) (10 pts) Fill in the information below. Also, fill in the A-Z header by

- **coloring in** the first letter of your first and last name (as it appears in Gauchospace),
- writing **either 4, 5, or 6** to indicate your **discussion section (lab)** meeting time
- 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:	
uemail address:	@uemail.ucsb.edu

If you collaborated with AT MOST one other person on this homework, write his/her name below. She/he should also have your name on his/her paper.

There is no new reading assignment for this homework--it continues the study of Chapter 16 from HFJ.

During W12, the CS56 students were assigned to create homework questions from Chapter 16. The remainder of the questions on this assignment are based on questions they came up with.

(2) (Thanks to Vicente C.) List, Set and Map are fundamental concepts in the Collections API of Java.

Briefly describe each of these (List, Set, and Map). For each, give an example of a programming situation where that one is more appropriate than the other two.

(a) (10 pts) List

(Continued on page 2)

CS56-W15-H17 page 2

(This question is continued from page 1)

(b) (10 pts) Set

(c) (10 pts) Map

(3) (10 pts) (Thanks to Raphael T.)

Bob is trying to implement a car class with subclasses Honda, Ford, and Chevy. He wants to create a method which can polymorphically take an ArrayList of Car, or an ArrayList of any of its subclasses (e.g. ArrayList<Ford>, ArrayList<Honda>).

He suggests the following method declaration:

```
public void takeCar(ArrayList<Car> list) { /* code goes here*/ }
```

What is wrong with Bob's logic and what method declaration would correctly implement what he's trying to do?