Name: *(as it would appear on official course roster)*		
Umail address:	**@umail.ucsb.edu**	section 4, 5, or 6
Optional: name you wish to be called if different from name above.		
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone"		

1

You may collaborate on this homework with AT MOST one person, an optional "homework buddy".

H12: Due Monday, 01.25 in Lecture

H12

CS56 W16

Review of (HFJ Ch1-8)
Assigned: Wed 01.20 Total Points: 50

MAY ONLY BE TURNED IN IN THE LECTURE/LAB LISTED ABOVE AS THE DUE DATE, OR IF APPLICABLE, SUBMITTED ON GRADESCOPE. There is NO MAKEUP for missed assignments; in place of that, we drop the five lowest scores (if you have zeros, those are the five lowest scores.)

There is no new reading assignment for this homework. Instead, this homework is review of material in Chapters 1-8 of HFJ.

---

1. (5 pts) Fill in the homework header properly—this helps us keep the grading pipeline flowing so that you get credit for your work and get feedback more quickly.
   - writing **either 4, 5, or 6** to indicate your **discussion section (lab)** meeting time
   - entering BOTH your name AND your umail address EVERY time.

   Paper submissions: One sheet of 8.5x11 paper double sided, or two DISCONNECTED SHEETS with your name on EACH. Please: **NO STAPLES, NO PAPERCLIPS, NO TAPE, NO ATTACHMENT OF ANY KIND**. These damage the document scanner.

   Scanned submission: When submitting by PDF upload: scan your pages legibly and **SCAN IN THE CORRECT ORDER**. Page 1 first, then Page 2, in the correct orientation. Failure to scan properly may result in zero credit, meaning you "use up" one of your five "drop the lowest grade" slots.

2. At right is some Java code. The main produces two lines of output.

   a. (5 pts) What is the first line of output? Circle one:

      a. `first=Phill last=Conrad`

      b. `first=Conrad last=Phill`

   b. (5 pts) What is the second line of output?

      a. `nums=[20, 30, 20]`

      b. `nums=[20, 30, 40]`

      c. `nums=[30, 20, 40]`

      d. `nums=[40, 20, 30]`

      e. `nums=[40, 30, 20]`

```
import java.util.ArrayList;
public class Prob2 {

    public static void swap(String a, String b) {
        String temp = a; a = b; b = temp;
    }

    public static void swap(ArrayList<String> a, int i, int j) {
        String temp = a.get(i); a.set(i,a.get(j)); a.set(j,temp);
    }

    public static void main(String [] args) {
        ArrayList<String>  nums = new ArrayList<String>();
        nums.add("20"); nums.add("30"); nums.add("40");

        String first = "Phill";
        String last = "Conrad";
        swap(first,last);
        swap(nums,0,2);

        System.out.println("first=" + first + " last=" + last);
        System.out.println("nums=" + nums);
    }

}
```

2

H12

CS56 W16

3. This homework exercise is the "sharpen your pencil" exercise from p. 169. The code is recreated at right. Note that in the real world, these three classes would be in three separate files, but to save space in the textbook, and on this homework, they are all in one listing.

   As far as I could tell, there isn't an answer key in the book anywhere for this exercise. But if you find one, good for you.

   In the questions that ask "how many instance variables", or "how many methods" does some class have, be sure to include any inherited instance variables and methods.

   Any when the question asks "Can a FamilyDoctor do treatPatient()?" the meaning is "can the treatPatient() method be invoked on an instance of class FamilyDoctor?"

```
public class Doctor {

    boolean worksAtHospital;

    void treatPatient() {
      // perform a checkup
    }
}
 public class FamilyDoctor extends Doctor {

    boolean makesHouseCalls;
    void giveAdvice() {
      // give homespun advice
    }
}
 public class Surgeon extends Doctor{

   void treatPatient() {
      // perform surgery
   }

   void makeIncision() {
      // make incision (yikes!)
   }
}
```

(a) (5 pts)	How many instance variables does Surgeon have?	
(b) (5 pts)	How many instance variables does FamilyDoctor have?	
(c) (5 pts)	How many methods does Doctor have?	
(d) (5 pts)	How many methods does Surgeon have?	
(e) (5 pts)	How many methods does FamilyDoctor have?	
(f) (5 pts)	Can a FamilyDoctor do treatPatient()?	
(g) (5 pts)	Can a FamilyDoctor do makeIncision()?	