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			

H18: Due Thursday, 02.12 in Lab

Garbage Collection, Polymorphism (Review for 2nd Midterm, HFJ 1-17)
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:	
umail address:	@umail.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. Instead, this homework is review for the second midterm, and refers to material from throughout Chapters 1-17 of the textbook.

---

(2) (10 pts) This exercise is a variation on the exercise on p. 266 in HFJ. The answer to that one appears in the book, so I suggest you try that one first, then check your answer before attempting this problem.

Along with this homework, there is a separate handout (if you don't have yours, consult the homework link on the wiki---the handout is in the same web page and is part of the same "printer friendly PDF".)

On the handout there is some code. Your job: figure out after which line of main() each of the following objects is eligible for garbage collection.

- (a) The dog named Fido?
- (b) The dog named Rover?
- (c) The dog named Princess?
- (d) The dog named Spot?
- (e) The dog named Snoopy?

If an object is still not eligible for garbage collection when the last line of main is reached, say "never". Your answer should be one of the line numbers that appears in comments in main (e.g. `/* 3 */` or `/* 7 */`).

(2) (Problem 12 from CS56 W12 Midterm 2) For this question, you need the additional handout with code for these files: Book.java, Product.java, Shippable.java, Song.java.

You may assume that all of the code on the handout compiles—I've checked that this is true. Now consider the following code, which does contain some errors, and as a result, will NOT compile.

```
/* 1*/   public class Q1 {
/* 2*/       public static void main (String [] args) {
/* 3*/
/* 4*/           Book gp = new Book("Pratchett","Going Postal",799,0.15);
/* 5*/           Song br = new Song("Lady Gaga","Bad Romance");
/* 6*/           Product slts = new Song("Nirvana","Smells Like Teen Spirit",79);
/* 7*/           Shippable hp = new Book("Rowling","Harry Potter & the Polymorphic Polyp",652,1.5);
/* 8*/           Shippable ttc = new Shippable("Dickens","Tale of Two Cities",999,1.5);
/* 9*/
/*10*/           System.out.println("a:" + gp.getTitle());
/*11*/           System.out.println("b:" + br.getArtist());
/*12*/           System.out.println("c:" + br.getPrice());
/*13*/           System.out.println("d:" + slts.getPrice());
/*14*/           System.out.println("e:" + slts.getTitle());
/*15*/           System.out.println("f:" + hp.getPrice());
/*16*/           System.out.println("g:" + hp.getWeight());
/*17*/           System.out.println("h:" + hp.getTitle());
/*18*/           System.out.println("i:" + ttc.getPrice());
/*19*/           System.out.println("j:" + ttc.getPrice());
/*20*/
/*21*/       } // main method
/*22*/   } // class Q1
```

Please do these things with this "broken" code for the Q1 class:

- (15 pts) Several lines need to be eliminated from this file in order to make it compile. Find the lines that are bogus, and draw a line through each of them in the code listing above.

Hint: By "several", I mean more than 2, and fewer than 10. Start by determining which, if any, of the constructors are bogus. Then, eliminate any lines that refer to the variables created on those lines. Finally, check all of the remaining method calls.

You will lose two points each time you striking a line that is not bogus, and you will lose two points for failing to strike any line that IS bogus. So, choose wisely.

- (15 pts) After striking through the bogus lines, the remaining code should compile and run. So, indicate what the output will be (if any) below. Be precise. If there will no output, write "no output".

The program for problem 2:

```
/** Exercise based on p. 266 in HFJ, for H16, S12, CS56, UCSB, P. Conrad
*/

public class Dog {

    private static Dog dogOfTheWeek = null;

    private String name;

    public void setAsDogOfTheWeek() {
        dogOfTheWeek = this;
    }

    public static Dog getDogOfTheWeek() {
        return dogOfTheWeek;
    }

    public Dog(String name) { this.name = name;}

    public static void main(String [] args) {

        Dog d1 = new Dog("Fido");
        Dog d2 = new Dog("Rover");
        Dog d3 = new Dog("Princess");
        Dog d4 = new Dog("Spot");
        Dog d5 = new Dog("Snoopy");

        d1.setAsDogOfTheWeek();   /*  1 */
        d1 = d2;                  /*  2 */
        Dog d6 = d3;              /*  3 */
        Dog temp = d4;            /*  4 */
        d4 = d3;                  /*  5 */
        d3 = temp;                /*  6 */
        d2 = getDogOfTheWeek();   /*  7 */
        d4.setAsDogOfTheWeek();   /*  8 */
        d5 = null;                /*  9 */
        d4 = null;                /* 10 */
        d3 = null;                /* 11 */
        d2 = null;                /* 12 */
        d1 = null;                /* 13 */
        temp = null;              /* 14 */
    }                             /* 15 */
}
```

Code for Problem 3:

Product.java

```
1   /** something that can be sold */
2   public abstract class Product {
3
4   /** get the price (in cents) */
5   public abstract int getPrice();
6
7   }
```

Shippable.java

```
1   /** something that can be shipped */
2   public interface Shippable {
3
4   /** get the shipping weight in pounds */
5   public double getWeight();
6   }
```

Book.java

```
1    /** A Book */
2    public class Book extends Product implements Shippable {
3
4    private int price;
5    private double weight;
6    private String author;
7    private String title;
8
9    public Book(String author, String title, int price,
10   double weight) {
11   this.author = author;
12   this.title = title;
13   this.price = price;
14   this.weight = weight;
15   }
16
17   public int getPrice() {return this.price;}
18   public String getTitle() {return this.title;}
19   public String getAuthor() {return this.author;}
20   public double getWeight() {return this.weight;}
21
22   }
```

Song.java

```
1    /** A downloadable Song */
2    public class Song extends Product {
3
4    private int price;
5    private String artist;
6    private String title;
7
8    public Song(String artist, String title, int price) {
9    this.artist = artist;
10   this.title = title;
11   this.price = price;
12   }
13
14   public Song(String artist, String title) {
15   this(artist,title,99);
16   }
17
18   public int getPrice() {return this.price;}
19   public String getTitle() {return this.title;}
20   public String getArtist() {return this.artist;}
21
22   }
```