**CS 180 Problem Solving and OO Programming**

Fall 2011
Recitation Week 5. September 19-23, 2011

**Problem 1:**

Point out errors, if any, in the program segments below.

(a)

```
public class Car{
      String make;

      public void getMake(){

            return make;

      }

}
```

(b)

```
public class Car{

      String make;

      public Car(){

            double speed=45.0;

      }

      public double getSpeed(){

            return speed;

      }

}
```

(c)

```
public class Car{

      String make;

      public static void main(String [] args){

            Scanner s=new Scanner(System.in);
```

```
            make=s.next();
        }
    }
```

(d)

```
    public class Car{
        String make;
        public static void main(String [] args){
            Scanner s=new Scanner(System.in);
            Car aCar=new Car();
            aCar.make=s.next();
        }
    }
```

(e)

```
    public class Car{
        static String make;
        public Car(String m){
            make=m;
        }
        public static void main(String [] args){
            Scanner s=new Scanner(System.in);
            Car aCar=new Car(s.next());
        }
    }
```

(f)

```
public class Car{

      static double speed=75.0;

      public static void main(String [] args){
            System.out.println(speed);

            Car aCar=new Car();

            System.out.println(aCar.speed);

            }

}
```

(g) In the `Math` class, which of the following is the likely declaration for `random()`? Why?

- `public double random();`
- `public static double random();`

**Problem 2:**

(a) We are required to write a Java program that reads a string of characters. It then extracts each character from the string and prints its ASCII code on the console. [Note: If `x` is a variable of type `char` then you my use the expression `(int)x` to obtain its integer equivalent which is its ASCII code.]

(b) Modify the above program so that it prints out the ASCII code of each character only if the code is less than 65.

Suggested steps to solve the problem:

1. Read the problem statement and understand what is required of the program. Resolve any ambiguities.
2. Design your program: What would you name the class? [Hint: Are we creating any objects?]
3. What methods should the class have? One method or more than one?
4. What are the inputs to each method and what is its output?
5. What should be the sequence of actions in the `main()` method?
6. Code your design by writing the classe designed and then compile and test.

<End of Problems for Week 5>