1. (2 pts) Suppose the following lines are in a main method? Will there be a compile-time error? If not, what is printed?

```java
int i;

i += 2;

System.out.println(i);
```

There will be a syntax error, because local variables must be initialized before they are used in a statement where their value is used.

2. (2 pts) Suppose the following lines are in a main method? Will 1 ever be printed? Why or why not?

```java
int i = 4;

while (i < 5) {
    System.out.println(i);
}

System.out.println(1);
```

No, 1 will never be printed. The while loop will continue to execute as long as the condition is true. Since there is nothing in the body of the while loop that changes the condition, the loop will execute forever, so the statement to print out 1 is never reached.

3. (2 pts) What is the major difference between a while or pre-test loop and a do-while or post-test loop? What does one guarantee that the other doesn’t?

In a standard while loop, the condition is checked before the body is executed. In a do-while loop, the body is executed before the condition is checked. So we are guaranteed in a do-while loop that the body will execute at least once.

4. (2 pts) Suppose we wanted to print out all even numbers between 0 and 1000. Show the code that will do this using a while loop.

```java
int counter=0;

while (counter <= 1000) {
    System.out.println(counter);
    counter += 2;
}
```
5. (2 pts) Recalling that the ASCII value of ‘A’ is 65, write code that uses a while loop that will generate two distinct random uppercase characters and then print those out.

```java
char firstChar = (char)(65 + (int)(26*Math.random()));

char secondChar = firstChar;

while (firstChar == secondChar)
    secondChar = (char)(65 + (int)(26*Math.random()));

System.out.println(firstChar + “ ” + secondChar);
```