CS 201 - Data Structures and Discrete Mathematics I – Spring 2005
Assignment 2: Recursion and Linked List

General Information

Deadline: 11:59pm, November 21, 2005 Worth: 100 points

The purpose of this homework is to practice linked list and recursion. You are required to use Java. The homework is not a group project and everybody should work on it individually. We will run your programs with MOSS, which gives us an indication whether two programs are too close to have been written independently.

Problems

1. (50 points) Given a sequence of numbers, write a class named **my_Special_Order** to store them in a linked list and then print them on the screen by starting from the middle, printing each number going left until beginning is reached, then starting from the middle and then printing each number (excluding the middle) going right until end is reached. For example, given "103 23 4 13 21 18 19" as input, your program should print out "13 4 23 103 21 18 19".

   Hint: Using the LinkedList Class, the new class should contain at least two methods: one for setting the sequence and pointers to beginning, middle and end, and one for printing the interior of the sequence in the required order.

2. (50 points) Write a java class named **my_Reverse** using recursion. It takes a string of letters or digits of length at most 100 and prints it in reverse order. For example, given "welcome 2005" as input, your program should print out "5002 emoclew". Note that space is the only separator between words.

Sample Run:

Problem 1:
Enter numbers:
1 2 3 4 53
The desired sequence:
3 2 1 4 53

Problem 2:
Enter string:
welcome 2005
Reverse string:
5002 emoclew

What to turn in

You should turn in two java files, one is named **my_Special_Order.java**, and the other is named **my_Reverse.java**. You should write main within each java file so that after compiling, we can run each of them separately.

How to turn in

login to bert.cs.uic.edu, go to your working directory and run **turnin**.

"turnin -c cs201 -p project1 my_Special_Order.java my_Reverse.java "

You may run **turnin** as many times as you want (only the last copy is saved).

You can use "turnin -c cs201 -p project1 –v" to check whether you have turned in successfully.

For more help, you can type

turnin -h
or
man turnin

Note

1. You MUST make sure that your program can compile using "javac" and run using "java" on bert.cs.uic.edu. For example, for the first problem, you should first compile using:

   javac my_Special_Order.java

   If no compiling error, you will see my_Special_Order.class created in your directory. Use

   java my_Special_Order

   to run your program.
2. You use an IDE, such as Jbuilder, to code and to debug, but you must make sure that your code can be compiled and run on bert.cs.uic.edu without any additional packages.
3. Remember to comment your code.
4. Zero mark will be given if your program does not compile, your program gets into an infinite loop (does not terminate) or you did not turn in a program.
5. You MUST use recursion for Problem 2. Only minimal credit will be given for any solution that does not use recursion even if your program works perfectly fine.