COMP 250 Winter 2009
Assignment 1
Instructor: Prof. Michael Langer
School of Computer Science
McGill University
Due: 6th Feb 2009, before 1:30 pm

General Information:

- Turn-in your code for all the questions electronically via webCT before class.
- The TAs who handle this assignment are Dorna Kashef, Boluan Pang, and Mathieu Rousseau.
- As you may know, there is a discussion board designed for this assignment in webCT which you can use for general discussion.
- **Late assignment policy:** Late assignment will be accepted up to only 3 days late and will be penalized by 20 points per day.

Question 1 (50 pts)

Background

In mathematics, a polynomial is an expression constructed from variables (also known as indeterminates) and constants, using the operations of addition, subtraction, multiplication, and constant non-negative whole number exponents. For example, $x^2 - 4x + 7$ is a polynomial.

Polynomials are one of the most important concepts in algebra and throughout mathematics and science. They are used to form polynomial equations, which encode a wide range of problems, from elementary word problems to complicated problems in the sciences; they are used to define polynomial functions, which appear in settings ranging from basic chemistry and physics to economics, and are used in calculus and numerical analysis to approximate other functions. Polynomials are used to construct polynomial rings, one of the most powerful concepts in algebra and algebraic geometry.

Question

Linked-lists will be used to represent polynomials for this question. You should use the given code (Poly.java) which defines polynomials in one variable with positive integer exponents and real coefficients. There is a class Term which represents a single term in the polynomial but also contains the link to the next term. Polynomials are always represented with a single term for each possible exponent, with the terms in decreasing order of the exponent and with the terms with zero coefficients removed from the representation. The class Poly contains a private Term inside it. Outside users of the package are meant to work with Poly rather than with Term. You should familiarize yourself with the code to understand how it works but the only changes you should need to make are in the Poly class in the methods:

addTerm	(20 points)
add	(10 points)
multiply	(10 points)

You may use the given main method to test if your solution is working but feel free to create your own test scenarios.

The clone method in the Poly class will help you with code reuse so make sure to take a look at how it works.

The final ten (10) points will be given for style (proper indentation, etc.) and useful comments so make sure to include these in your solution as well.

What to submit: Please upload your "Poly.java" file to WebCT directly.

Question 2 (50 pts)

The purpose of this question is to practice the concept of inheritance and polymorphism. You are provided with 3 files named: Person.java, Employee.java, and Manager.java. You should familiarize yourself with the code to understand how it works.

- Person class has attributes: name, age, gender, and savings.
- Employee is-a-kind-of Person which inherits the attributes of Person, and also have extra attributes: ID, employer, and salary.
- Manager is-a-kind-of Employee which inherits the attributes of Employee, and also have extra attributes: department, and bonus.

1. (10 pts)

Employee class has a method named *getPay* which returns the base salary of the employee. Manager class inherits this method; however, each manager has a bonus which should be added to his/her base salary. In the Manager class, override *getPay* method so that it returns the updated (gross) salary.

2. (15 pts)

The sales department of the company has several sales people. Write a class in order to keep the information about each sales person. The class should be named "SalesPerson.java". Note that sales person is-a-kind-of employee; therefore, SalesPerson should be a subclass of Employee.

SalesPerson class should have two extra attributes: total sales, and a commission rate. Total sales will be a float value indicating the total value of sales (dollars and cents). The commission will be a percentage indicating the share of the sales that will be paid to the sales person. Your method should override the *getPay* method, making it return the base salary plus the commission (defined as the product of the total sales and the commission rate). You should also override *toString* method (Hint: Take a look at the Manager class to see how this method should be overridden.).

3. (25 pts)

Write a class named "Company.java". This class is responsible for maintaining a list of all employees, and can act as the main class to test your project.
You should use only **one** linked list to store the employees.
Your class should have the following methods: (You can also add more methods if needed.)

1. **public void addEmployee(Employee e)**
   This method adds the employee *e* to the linked list of employees.
2. **public void findMaxSalary()**
   This method finds the employee with the highest gross salary (which includes bonus for managers or commission for sales people), and prints out his/her information (name, age, gender, savings, base salary…). Make sure to also print out the gross salary.
   public static void main(String args[])
   In this method, fill the employees linked list with the list of employees provided in the table below by using *addEmployee* method. Call *findMaxSalary* afterwards to print out the information about the employee with the highest salary.

Name	Gender	Age	Savings	Position	ID	Employer	Base Salary	Department	Bonus	Total Sales	Commission Rate
Mary	F	35	200,000	Manager	1	Simons	80,000	Technical	1200	_	_
Jack	M	42	15000	Manager	2	The Bay	90,000	Human Resource	800	_	_
Catherine	F	33	9000	Sales	3	The Bay	60,000	_	_	140,000	0.18
Lincoln	M	27	5000	Sales	4	The Bay	48,000	_	_	332,000	0.3
Lin	F	51	30000	Staff*	5	The Bay	45,000	_	_	_	_
Mike	M	44	10000	Staff	6	Simons	51,000	_	_	_	_
Sharon	F	42	5000	Staff	7	Simons	58,000	_	_	_	_
Jessica	F	25	1000	Staff	8	Simons	38,000	_	_	_	_
Calvin	M	38	12000	Staff	9	The Bay	43,000	_	_	_	_
Berta	F	22	3000	Staff	10	The Bay	30,000	_	_	_	_

* Staff is a basic employee.

You are also provided with a text file named "employee_information.txt" which you can use to copy/paste the above information into your code.

What to submit: Please upload Manager.java, SalesPerson.java, Company.java (and other java files that you have modified, if any) directly to WebCT.

Your code will be evaluated on several criteria such as correctness, reuse of available code, style and comments.