CM10277: Principles of Programming I Coursework
Part 3: The Java Programs

Dr. Marina De Vos
Dept. of Computer Science
University of Bath
mdv@cs.bath.ac.uk

Contents

1 Introduction

The coursework of this unit consists of three parts: the Lab, Python and the Java coursework. This document provides the specification for the final part: Java.

Although you can use any IDE to develop your code, we expect the programs to be executable using JDK from the command-line (i.e. we have to be able to run it use javac and java) without requiring the installation of libraries, modules or other programmes.

Questions regarding the coursework can always be posted on the Moodle Forum and programming1@lists.bath.ac.uk mailing list.

2 Learning Objectives

At the end of this part of the coursework you will be able to design and write medium-sized programs containing a small number of classes by using the appropriate object-oriented software techniques of data encapsulation, inheritance and polymorphism.

3 The Program: Calculator

Write a program that takes as input a mathematical expression consisting of decimal numbers and the four basic mathematical operators (addition, subtraction, multiplication, division) and nested mathematical expressions, and evaluates this expression.
Some examples:

- 5 + 6 * 2 results in 17
- (5 + 4) * (4 - 2) results in 18
- (4 + (2 * (2 - 1))) * 1.2 results in 7.2
- -2 / 4 results in -0.5

Hint: Use the Composite Design pattern.

Include with your program a file name Extension.txt in which you explain how your system can be extended to include other mathematical operators.

A complete implementation of the above system with a text-based interface is worth 80% with 25 marks for running the system with correct input, 10 marks for robustness, 35 marks for the program structure and 10 marks for the discussion on extendibility.

The remaining 20% is set aside for the implementation of a graphical user interface for this application. and the capability of storing results for further computations.

4 Assessment

4.1 Conditions

The coursework will be conducted individually. **Attention is drawn to the University rules on plagiarism on page 60-62 of the Student Handbook.** While software reuse (with referencing the source) is permitted, we will only be able to assess your contribution.

4.2 Estimated Workload

All coursework combined contributes 50% to unit's mark. This part of the coursework is worth 20% of your final mark.

This unit is a double unit, so in theory you should spend 200 hours on it. Lectures and preparing for the exam (e.g. doing the reading, revising your notes) should take 72 hours. Since the java coursework is worth two-fifth of the remaining hours, you might expect to spend about 50 hours.

However, programming is a skill. It requires both knowledge and practice. How easy you will find it depends also partly on your previous experience and abilities. Different students will spend very different amounts of time on these lab sheets. Further, the same student may find some programming concepts easy and others difficult, and which parts seem more difficult will vary from student to student. We suggest you allocate as much time to programs as they take you. However, if you are at all confused or feel you are not making progress, be sure to ask for help either in lab, on Moodle or from the mailing list.

4.3 Marking

Key issues for marking the code will be: compiling, running with expected input, robustness (handling incorrect user input), module design, proper use of data encapsulation and decomposition, and the algorithms being used.

In cases where it is not clear from the assignment how it should be marked, you may be called on to explain or demonstrate your program. Such cases include but are not limited to suspected plagiarism.

5 Submission Instructions

The deadline for this part of coursework is **10am Tuesday 08 January 2013**. Before the deadline upload your solution via Moodle (`http://moodle.bath.ac.uk/moodle5/course/view.php?id=30475`).

Your program solution should be a zip or tar.gz file that can be extracted to a directory with the name yourusernameCalculator. The directory should contain at least a class Calculator.java which contains the

main method to run your program. Before you upload your solution, make sure that the zip or tar.gz file contains all necessary files and creates the correct directory.

Please download the file from moodle and double check that you have attached the right file, with the content that you want to be marked. You are responsible for checking that you are submitting the correct material to the respective assignment.

6 Feedback

An opportunity for individual detailed feedback will be provided either via Moodle or a feedback session. Marks will be made available Moodle within three weeks of the submission deadline. Clarification can be obtained from the unit lecturer.