

CM50147: Programming I Coursework

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

Contents

1	Introduction	1
2	Objectives	2
3	Questions	2
3.1	Program 1 (20 points)	2
3.2	Program 2 (25 points)	2
3.3	Program 3 (25 points)	2
3.4	Essay (30 points)	3
3.4.1	Format	3
3.4.2	Topic	3
4	Assessment	3
4.1	Conditions	3
4.2	Estimated Workload	4
4.3	Marking	4
5	Deadline	4

1 Introduction

The goal of this coursework is two-fold. The first part consists of designing and implementing three medium-sized programs. Although you can use BlueJ as a tool, we expect that the programs are fully running using JDK and the command-line. The second part of the coursework consists of writing a scientific essay.

Questions regarding the coursework can always be posted on the `programming1@bath.ac.uk` and `tutors-programming1@cs.bath.ac.uk` mailing lists.

2 Objectives

At the end of this 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. Furthermore, you will have demonstrated reflection on programming concepts and appreciation of the different approaches to design used in empirical and engineering oriented sciences.

3 Questions

3.1 Program 1 (20 points)

Design and implement an inheritance hierarchy to model the animal kingdom (i.e. the animal taxonomy). There is no need to implement the full taxonomy. Only that part of the taxonomy necessary to populate your hierarchy with ten different animals is required. Your sample should at least include one reptile, one mammal, one fish and one amphibian. Write a driver program that asks the user for how many animals (s)he wants in his/her zoo. Use this input to randomly create the necessary animals and store them in a container class. Write methods to allow the zookeeper to add/delete random animals and to take stock.

Answer in the comments of your driver class why asking a list of all mammals would be a breach of polymorphism.

3.2 Program 2 (25 points)

Rewrite question 1 such that instead of an inheritance hierarchy you use the composite design pattern for the representation of the animal kingdom.

Answer in the comments of your driver class if this does this solve the breach of polymorphism? Why (not)? Explain which approach you would choose.

3.3 Program 3 (25 points)

Write a simulation for a phone company. The company serves two types of clients depending whether they use a mobile phone or not. When you want to make a call you ask the phone company for a connection. In return you receive a unique callID. This will be used to maintain the connection. The person called will obtain the same callID. In order to do so the phone company will contain a collection of connections, a collection of phones. Whenever a connection is requested an object of type Connection is created containing the caller and the callee. The communication between the two parties is established by passing messages to the phone company who will pass it on to the corresponding Connection for handling. Charges for the call are based on the type of phones used and the number of messages sent to a Connection. A connection can be closed by either party after which the caller is informed of the cost of the call. Make sure that you provide means to track the connections and phone calls. Provide a program that demonstrates the full capabilities of your system.

One of the main aims and marking criteria of this question is the design of your system. We will take into account the usage of abstract classes, inheritance and polymorphism. A good and robust demonstration of your program is also one of the marking criteria.

NOTE: Some of this question description is left intentionally vague to stimulate your own creativity.

HINT: Have a closer look at the LAN-simulation we have done during the lab-sessions.

DESIGN: The design of your program can be commented on by the tutors. However, this will cost you **5 marks** for this question. This should not stop you from asking general design questions. Tutors will be before

answering a 5 point question inform you about this.

3.4 Essay (30 points)

3.4.1 Format

1. An essay (hereafter, "paper") of 3000 words in length, including title and abstract, but not including references (i.e. the length of the reference section shall not be deemed to contribute to the total word length of the paper).
2. Diagrams may be included if helpful to the argument being made in the paper.
3. The paper must be formatted and sectionalized in accordance with the stipulations of the ACM Proceedings Format. (<http://www.acm.org/sigs/pubs/proceed/template.html>).
4. The paper must have an original title, an abstract of no more than 150 words, and a reference section.
5. Papers that fail to meet the word length specification¹ will be penalised according to the following tariff:
 - 0 - 100 words, no penalty
 - 101 - 250 words, deduction of 10%
 - 251 - 500 words, deduction of 20%
 - 501 or more words, deduction of 50%
6. The paper will be marked by Dr. Leon Watts, Director of Studies for the HCC programme.

3.4.2 Topic

In every almost every sphere of human activity, methods have been devised to assure the validity and verifiability of the things that people do. Amongst other matters, the MSc in Human Communication and Computing is concerned with design activities carried out by software engineers (including modular decomposition of a problem and programming) and by empirical investigators (including definition of observational phenomena and data gathering). How does the idea of design in programming compare to the idea of design in an empirical study? You should pay attention to the concepts of procedure, dependency, variable and factor as they are used in each sense. You should consider whether there is an equivalent to a set of hypotheses in the design and implementation of a computer program. You should also consider whether there are equivalents of programming objects in the design and prosecution of an empirical study.

4 Assessment

4.1 Conditions

The coursework will be conducted individually. Attention is drawn to the University rules on plagiarism on page 49 of the student handbook. There are 4 questions and you should answer all of them. You will only be entitled to marks for this coursework if you have successfully and timely completed 7 of the labs.

¹Conferences and journals all have a standard publication format and submission guidelines. Submissions that do not adhere to the specification are routinely rejected out of hand. Typically, submissions that do not adhere to specifications are evidence of shoddy preparation or muddles thinking.

4.2 Estimated Workload

You need 40% to pass this part of the unit. The average amount of time you are intended to spend on this coursework is 30 hours.

4.3 Marking

The first question is marked out of 20. The two other programming questions constitute for 25 points each. The essay is worth 30 marks. If you want to pass, you should attempt all four questions. Key issues for the programming component will be: compiling, running with expected input, robustness (handling wrong user input), class and code design, proper use of inheritance and polymorphism, and the algorithms being used. For the essay marks can be obtained: for coherence, structure, well-stated opinions, proper use of reference and a to-the-point abstract.

5 Deadline

The deadline for the coursework is **January 13th 2004 5pm (17.00)**. Before or on the day of the deadline, hand in, at the department office, samples of input and output of your programs, a floppy/CD-ROM containing the source code of each of the questions in a separate directory and a print-out of your essay. Name the class with the main-method MainX with X the number of the question. Make sure that the code on the floppy is compiling and running as it should (`javac *.java` and `java MainX` from the appropriate directory). Renaming errors, wrong versions and bad disks will **not** be accepted as an excuse.

It is highly advised that you also include a print-out of your code in case the electronic version of your code turns out be faulty.