

CMP1024 Further Programming
Assignment One
Subject to Moderation

Stephen Rank,
Department of Computing and Informatics,
Faculty of Technology,
University of Lincoln
srank@lincoln.ac.uk

March 8, 2007

1 Introduction

This assignment draws on all the material covered in the unit so far, and allows you to demonstrate your understanding of both OO programming and OO design.

You should hand in your program on CD and paper as well as a report, following the outline of the specification below.

The Learning Outcomes addressed by this assignment are:

LO1 explain the object-oriented approach in simple terms;

LO3 create and instantiate classes in an object-oriented programming language;

2 Analysis and Design

In a computer game, clans are made up of a set of players. Each clan has a name, a flag, a store containing clan items and some cash. Each player has a name, an experience score, and a collection of items (weapons, armour, *etc*). Produce a UML class diagram showing these classes and their relationships.

3 Programming Tasks

This part of the assignment is based on writing and testing a small class to represent dates.

Your report should contain an annotated copy of the source code, along with description of the steps that you used to test your program.

- Write a class declaration for `Date`, representing a `Date` as day, month, and year. Use appropriate access control.
- Write a method `void set(int d, int m, int y)` which sets a `Date` to a particular value.
- Write a member function called `output` which displays the value of a `Date` in suitable form.
- Write member functions to add and subtract years and months from a `Date`
- Write member functions `tomorrow()` and `yesterday()` that add or subtract one day from a `Date`. Take care to handle the beginning and end of months and years correctly.
- Write a member function that converts a `Date` into the number of days before/after the first of January 1900. Use this function to calculate your age in days.

Note that your marks for this section will also depend on the quality of your code, in terms of comments, layout, naming conventions, and other stylistic points.

Learning Outcome	Criterion	Pass	2ii	2i	First
Explain the object-oriented approach in simple terms.	Develop a class model for a simple system.	Each class correctly identified, with some correct operations and attributes.	Whole diagram uses correct UML syntax. Appropriate relationships defined.	All attributes and operations correct, with evidence of sound justification.	Account taken of users' needs, possibility of changes, or other advanced features.
Create and instantiate classes in an object-oriented programming language.	Produce and test a Date class in Java.	Syntactically correct program.	Class correctly modelled in terms of attributes and operations, with some correct implementation.	Good, well-tested and well-thought-out methods and attributes with evidence of testing and evaluation.	Thorough evaluation of the system, with evidence of analysis of the problem and potential uses of the program.

4 Criterion Reference Grid