CM50258: Principles of Programming
Coursework 2: Java Connect4

Dept. of Computer Science
University of Bath

1 Introduction

This document provides the specification for the second Principles of Programming coursework: **Connect4**.

Questions regarding the coursework can be posted on the **Moodle Forum**, sent via the **programming1@lists.bath.ac.uk** mailing list, or asked to the lecturers/tutors during the Lecture or Lab sessions. **Please do not post part or the whole of your code solution on the Moodle forum.**

2 Learning Objectives

At the end of this coursework you will be able to

- Plan, organise and implement program code to support reuse and maintainability of a software project.
- Provide a critical review of a software artefact in terms of software quality, design, reuse and robustness, and offer solutions to correct issues encountered.

3 Connect4

For your first coursework, we asked you to investigate and replicate a piece of code that we supplied (SRPN). In this second piece of coursework you will need to analyse, design and extend a second program which you can find on Moodle (Connect4).

The challenges in this coursework arise from the fact that the Java code supplied should allow a human to play the game Connect4 against a computer. Connect4 is described at https://en.wikipedia.org/wiki/Connect_Four. Please ignore the "Rule variations" listed on Wikipedia. You should only implement the standard version of the game.

Unfortunately, our given code:

- **doesn't compile or run,**
- **is poorly designed (e.g. It is mostly made up of a huge main function without comments)** and
- **needs to be fixed while using the object-oriented principles/features**.

To resolve these problems, and therefore to gain marks on this coursework, you will need to satisfy Requirements 1-3 below.

4 Coursework2 Requirements

1. Provide a bug and omission list (1-2 pages of single-spaced, 10 point font) explaining why our version of the code doesn't work (in the case of bugs) and/or doesn't provide the functionality needed to play Connect4. Each bug/omission on the list should be described in the following format:

   - Class : Line Number(s) : Bug/Omission : Bug Type if Bug (syntax/run time/logic)
   - Solution.

   For example:

   - Board.java : Line 7 : char entered where int expected : syntax error
   - Solution: Guard against non integer inputs.

2. Submit a revised version of our code which compiles and runs to provide a working version of the Connect4 game. More specifically, provide an altered version of our code which a human marker can use to play a complete game of Connect4 against the computer. You may wish to start by commenting out large parts of our code and altering our printBoard() method. This approach has the advantage of giving you a relatively manageable starting point in the debugging process. You may also wish to tackle our placeCounter() method next.

3. Provide comments within your written code to describe your choice of classes and solution. Description of solution includes the description on the use of object-oriented programming principles/features (such as encapsulation, modularisation, abstraction, inheritance, interfaces e.t.c).

5 Assessment

CM50258 is made up of 50% exam and 50% coursework. The Coursework is divided into two parts where 20% is from the SRPN coursework, and the remaining 30% from the Connect4 coursework. Note that the work described in this document therefore counts towards 30% of the course total.

If you have any questions about anything in this document, please do not hesitate to ask us.

The coursework will be conducted individually. Attention is drawn to the University rules on plagiarism. Whilst reference to existing code (with appropriate citation of that source) is permitted, we will only be able to assess your contribution.

You may be called on to explain or demonstrate your program. Cases leading to such an explanation and/or demonstration include but are not limited to suspected plagiarism.

6 Mark Scheme

Marks for this coursework (Connect4) are available in the following areas:

1. Bug/Omission list: Max 30 marks.
2. Revised (i.e. working) Connect4 code: Max 45 marks.
3. Comments on choice of classes and solution: Max 25 marks

Total: Max 100 marks.

The marking will be carried out using a rubric based on the above criteria. For the code that you submit, marks will be awarded for functionality, code quality and appropriate use of object oriented programming principles.

7 Submission Instructions

The deadline for this coursework is at **8pm on 17th December 2021**. You must upload your submission before the deadline. Please leave adequate time for checking that your code compiles and runs; also ensure that you have your comments for the choice of classes and solution within the code.

Your submissions must be a zip file that can be extracted to a directory with the name: Connect4-yourusername. The directory should contain:

- a bug/omission list,

- a sub-directory containing the Java files required to run Connect4.

Your .zip file must be submitted to the Moodle assignment page by the submission deadline shown. Submissions received after this deadline will be capped at 40% if received within 5 working days. Any submissions received after 5 working days will be marked at 0%. If you have a valid reason for an extension, you must submit an extension request through your Director of Studies – unit leaders cannot grant extensions.

You should leave yourself time to download your file from Moodle, extract it, and check that you have attached the correct file, with the content that you want to be marked. **You** are responsible for checking that you are submitting the correct material to the correct assignment.