Coursework: Collaborative Software Development

Module: Professional Skills (COMP 51915)
Term: Michaelmas Term 2025
Lecturer: Christopher Marcotte

Submission Please submit a `zip` archive of your repository and peer-evaluation form.

Deadlines Consult the MISCADA learning and teaching handbook for submission deadlines.

Plagiarism and collusion Students suspected of plagiarism, either of published work or work from unpublished sources, including the work of other students, or of collusion will be dealt with according to Computer Science and University guidelines.

Important Information You will have received a *Peer Assessment* form with this assignment, which you should include in your submission, see 2. Likewise *if there are any issues with your group peers which you expect will affect the quality of your submission*, you should contact me early for intervention.

You are expected to work together in a group using the collaborative software development skills discussed in the workshop. Any submission content which is produced entirely offline will be considered to **not demonstrate the learning outcomes***. Your repository should remain private to your group.*

This assignment is group work. You should work collaboratively in a shared repository based on the template repository.

1 Instructions

For this coursework, you are a new employee of a game-design company designing the next big multi-player board game for the burgeoning genre of computer-assisted offline games, marketed as *Cybergames*. You are to design a new game taking place on an $M \times N$ grid (e.g. $3 \leq M, N \leq 129$), and each position (grid cell face) permits a player index $p \in \{1, 2, \ldots, P\}$, like a generalisation of Tic-Tac-Toe on a larger grid and played with P players. You must design the win conditions for your game − what pattern each of the 1 through P players should seek to create to win the game.

For the coursework, you are tasked to:

- organize yourselves into groups;
- create a private repository for your group project based on the template repository;
- list your collaborators and name your game in the `README.md`;
- define your game and the win conditions in the `README.md`;
- finish the `parseBoard` function by parsing the board state;
- implement tests using your preferred testing framework;

- create a build system and give instructions in the `README.md`; and
- implement continuous integration (CI) for your software repository.

1.1 Testing Frameworks

There are a number of simple testing frameworks for C/C++. You should select one and use it consistently for all your testing. Several suggestions are collected below (from least complex to most complex):

- https://nemequ.github.io/munit
- https://github.com/mity/acutest
- https://github.com/google/googletest
- https://www.boost.org/

Suggestions It is expected that you should spend between 4 to 6 hours on this coursework (i.e. one or two evenings); I recommend you work diligently and **stop** at the 6 hour mark.

You are permitted to modify the template code provided, and to make the underlying game software as complex as you prefer. However, you should keep in mind the marking criteria.

2 Submission

You will submit a *zip archive* of your repository directory (`repo/`) and the peer submission form `peer-evaluation-form.docx`. You can form your submission file with the command:

```
zip -r <studentID>.zip repo peer-evaluation-form.docx
```

You would then submit the `<studentID>.zip` file on BlackBoard Ultra, with **your student ID** (e.g. `abcd12`) in place of `<studentID>`.

Your submission will include a peer-evaluation form for your group peers, which accompanies this document. It is essential that you fairly assess your peers and their contributions to the project, and that you submit one peer-evaluation form for your group (not including yourself). Inappropriate or unprofessional content in the forms will be marked down.

3 Marking Criteria

All submissions will be marked using the following criteria:

Marking Category	Marks (/100)
Effective use of Version Control features and commands at all stages.	20
Functional build system and clear build instructions.	20
Thorough testing of the program and win condition logic.	20
Functional Continuous Integration with appropriate running.	20
Peer-evaluation form is submitted, complete, and appropriate.	20

Table 1: Marking criteria and mark weighting for the coursework.

Generic coursework remarks

Stick exactly to the submission format as specified. If you alter the format (submit an archive instead of plain files, use Word documents rather than PDFs, …), the marker may refuse to mark the whole submission. Markers will not ask for missing files. If you have to submit code, ensure that this code does compile and, unless specified otherwise, does not require any manual interaction. Notably, markers will not debug your code, change parameters, or assess lines that are commented out.

All of MISCADA's deadlines are hard deadlines: In accordance with University procedures, submissions that are up to 5 working days late will be subject to a cap of the module pass mark. Later submissions will receive a mark of zero. If you require an extension, please submit an official extension request including medical evidence and/or acknowledgement by college. Do not contact the lecturers directly, as lecturers are not entitled to grant extensions. Details on extensions and valid reasons to grant extended deadlines can be found in the Learning and Teaching Handbook.

It is the responsibility of the student to ensure that there are sufficient backups of their work and that coursework is submitted with sufficient slack. Submit your coursework ahead of time. If in doubt, submit early versions. Technical difficulties (slow internet connection around submission deadline, lost computer hardware, accidentally deleted files, …) will not be mitigated. Please see `https://www.dur.ac.uk/learningandteaching.handbook/6/2/6/` for further information regarding illness and adverse circumstances affecting your academic performance.

If collusion or plagiarism is detected, both students who copy and students who help to copy can be penalised. Do not share any coursework with other students, do not assist other students, cite all used sources incl. figures, code snippets, equations, …Please see `https://www.dur.ac.uk/learningandteaching.handbook/6/2/4` and `https://www.dur.ac.uk/learningandteaching.handbook/6/2/4/1` for further information.

Coursework is to be treated as private and confidential. Do not publish the whole or parts of the coursework publicly. This includes both solutions and the plain coursework as handed out.

Group work

This work is group work. You will be assigned to small groups of students to complete this assessment prior to the last lecture of the course. Students can not change the assignment to groups or request to be made member of a particular group under normal circumstances.

Every group member has to submit individually. If you are asked to work collaboratively on one piece of work, every team member has to submit a copy of the work. Please ensure that the work clearly identifies your group members, i.e. enlists all collaborators.

While the work is completed as a group, you will receive individual marks depending on your individual performance.

Extension requests have to be submitted individually and do not carry over to all team members. If individual group members cannot make significant contributions, the marking will take this into account as contextual factor for the affected members of the group. For these reasons, it is particularly important to complete collaborative group work ahead of deadlines, and to clearly keep track and provide evidence for affected days if unexpected circumstances arise.