Cardiff School of Computer Science and Informatics
Coursework Assessment Pro-forma

Module Code:	CMT304
Module Title:	Programming Paradigms
Lecturer:	Víctor Gutiérrez-Basulto
Assessment Title:	Logic Programming
Assessment Number:	1 of 4
Date Set:	31st October 2022
Submission date and Time:	by 5th May 2023 at 9:30am
Feedback Return Date:	2nd June 2023

The submission deadline for *extensions* due to extenuating circumstances will be two weeks after the submission date above.
Feedback of coursework with extensions will be returned two weeks after the feedback return date above.

This is assignment **one** of a portfolio that will be composed of **four** assignments. This assignment is worth 25% of the total marks available for this module.

If coursework is submitted late (and where there are no extenuating circumstances):

1. If the assessment is submitted no later than 24 hours after the deadline, the mark for the assessment will be capped at the minimum pass mark;

2. If the assessment is submitted more than 24 hours after the deadline, a mark of 0 will be given for the assessment.

Extenuating circumstances (extension or deferral) can *only* be requested using the extenuating circumstances procedure: https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/extenuating-circumstances. Only students with *approved* extensions may use the extenuating circumstances submission deadline – you will receive an *approval* e-mail; this is not the e-mail confirming the *submission*. Any coursework submitted after the initial submission deadline without *approved* extenuating circumstances will be treated as late. Note, if you apply for *deferral*, instead of an extension, you will be given the opportunity of a *reassessment* at the next opportunity. You can apply for deferral after an extension request if this is before the (extended) deadline. More information on the extenuating circumstances procedure can be found on the Intranet: https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/extenuating-circumstances.

By submitting this assignment you are accepting the terms of the following declaration:

I hereby declare that my submission is all my own work, that it has not previously been submitted for assessment and that I have not knowingly allowed it to be copied by another student. I understand that deceiving or attempting to deceive examiners by passing off the work of another writer, as one's own is plagiarism. I also understand that plagiarising another's work or knowingly allowing another student to plagiarise from my work is against the University regulations and that doing so will result in loss of marks and possible disciplinary proceedings[1].

---

[1] https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/academic-integrity/cheating-and-academic-misconduct

Assignment

Consider the problem of solving Hitori puzzles. Hitori is played in an $n \times n$ grid of cells, where each cell contains a number $j \in \{1, \ldots, n\}$. The objective is to remove cells (marked as black), so that the three following rules are satisfied.

1. No number appears more than once in a row or column.

2. The eliminated black cells cannot be horizontally or vertically adjacent.

3. The remaining numbered (white) cells must be connected to each other, i.e., any white cell is reachable from another white cell by vertical or horizontal movements through white cells.

You can find an example of a Hitori puzzle and a solution in Wikipedia: https://en.wikipedia.org/wiki/Hitori

Task 1: Write an ASP program (`hitori.lp`) that solves the puzzle for any instance. Your program will receive as input a set `cell/3` of triples r, c, n such that the grid has number n in row r, column c. The output of your program is a set `cellblack/2` of pairs r, c such that the cell at row r, column c is eliminated.

Document your code so the following is clear.

1. What the approach to solving the problem is. In particular, you need to explain **what** each rule achieves and **how** the rule achieves it.

Include your name and student id in the comments.

Task 2: Write a short report on answer set programming related to the problem:

1. Provide, in up to 300 words, two arguments for and two arguments against using answer set programming to solve *this problem*.

2. Discuss, in up to 300 words, whether the logic programming paradigm is suitable for *this problem* or whether another paradigm of your choice is more appropriate, based on your previous arguments.

The word limits are an upper limit, not a target length. Text longer than the word limit for each point may be ignored. Clearly mark each argument in your answer of the first point and indicate whether it is for or against. Only provide two arguments for and against; additional arguments will be ignored.

Learning Outcomes Assessed

- Explain the conceptual foundations, evaluate and apply various programming paradigms, such as logic, functional, scripting, filter-based programming, pattern matching and quantum computing, to solve practical problems.

- Discuss and contrast the issues, features, design and concepts of a range of programming paradigms and languages to be able to select a suitable programming paradigm to solve a problem.

Criteria for assessment

Task 1: maximum 50 marks, assessed according to the following scale:

Fail	0	No code has been submitted.
	$1-14$	Code does not run or does not produce valid output for any valid input; little to no relevant documentation.
	$15-24$	Code is valid without syntax errors and creates a valid output for every valid input (or produces a suitable error message for valid cases it cannot process). The output is not a solution, but a suitable attempt to solve the problem is visible. An attempt to document the code has been made.
Pass	$25-29$	Code is valid without syntax errors and creates a valid output for every valid input (or produces a suitable error message for valid cases it cannot process). A suitable attempt to solve the problem has been made, that will often produce the correct output. The attempt has been reasonably documented.
Merit	$30-34$	Code is valid without syntax errors and creates a valid output for every valid input (or produces a suitable error message for valid cases it cannot process). A suitable attempt to solve the problem has been made, that will find the correct output. The attempt has been well documented, stating the idea to solve the problem and how it has been implemented.
Distinction	$35-50$	Code is valid without syntax errors and creates a valid output for every valid input. A suitable attempt to solve the problem has been made, that will find the correct output for all problems, with excellent performance. The attempt has been well documented clearly stating the idea to solve the problem and how it has been implemented. It clearly shows an effort to properly use the main features of the logic programming paradigm.

Task 2: maximum 50 marks, assessed according to the following scale:

Fail	0	No document has been submitted.
	$1-14$	An insufficient number of arguments has been submitted and/or they hardly apply to the logic programming paradigm. At most an incomplete attempt to discuss the suitability of the logic paradigm has been made.
	$15-24$	An insufficient number of arguments has been submitted, but they show some understanding of the logic programming paradigm. An attempt has been made to discuss the suitability of the logic paradigm, but it hardly relates to the paradigm.
Pass	$25-29$	The required number of valid arguments has been submitted. They are generally valid for the logic programming paradigm, but they repeat similar issues, do not consider the specific problem or contain mistakes in the details. A attempt has been made to discuss the suitability of the logic paradigm and some understanding of this paradigm is present.
Merit	$30-34$	The required number of valid arguments has been submitted. They show a clear understanding of the logic programming paradigm and it relates to the problem. The discussion of the suitability of the logic paradigm is well-developed, showing a clear understanding of the issues involved, and indicates the differences to the other chosen paradigm.

Distinction	$35 - 50$	The required number of valid arguments has been submitted. They show a clear understanding of the logic programming paradigm and the underlying theoretical concepts and/or realisations on programmable machines and how these relate to the problem. The discussion of the suitability of the logic paradigm is well-developed, showing a deep understanding of practical and theoretical issues involved, and clearly discusses concrete differences to the other chosen paradigm.

Feedback and suggestion for future learning

Feedback on your coursework will address the above criteria. Feedback and marks will be returned on 2nd June 2023 via Learning Central. This will be supplemented with individual feedback on request via e-mail.

Submission Instructions

All submissions must be via Learning Central. Upload the following files in a **single zip file**, `[student number].zip`:

Description		Type	Name
Task 1	**Compulsory**	One source file	`hitori.lp`
Task 2	**Compulsory**	One PDF (.pdf) file	`task2.pdf`

Any code submitted will be run on a system equivalent to the Linux laboratory machines and must be submitted as stipulated in the instructions above.

Any deviation from the submission instructions above (including the number and types of files submitted) may result in a mark of zero for the assessment or question part.

Staff reserve the right to invite students to a meeting to discuss coursework submissions.

Support for Assessment

Questions about the assessment can be asked on https://stackoverflow.com/c/comsc/ and should be tagged CMT304; or you can ask your question in or after any synchronous session with the assessment setter.