



# Course Directive

## IN628: Programming 4

### Semester One, 2020

## Description

In this paper, students will be introduced to non-anthropomorphic object-oriented class architectures and a selection of associated core computational approaches. This paper is the fourth semester of the programming thread and will be taught in a range of industrially relevant languages. The problem context may include games, modelling and simulation, artificial intelligence and other areas of appropriate computational and informational complexity.

## Course Information

Credits: 15 Credits  
Prerequisite: IN610: Programming 3  
Timetable: Wednesday 3pm D207 & Friday 1pm D105b

## Lecturer

Name: Grayson Orr (Lecturer)  
Location: D311  
Email: grayson.orr@op.ac.nz

## Course Dates

Term 1: 17 February - 25 March (6 weeks)  
Mid Semester Break: 26 March - 14 April  
Term 2: 15 April - 19 June (10 weeks)

## Aims

To expose students to a wide range of programming languages and paradigms, and continue their development of understanding of algorithms, architecture, and data structures.

## Learning Outcomes

At the successful completion of this course, students will be able to:

1. Program effectively in an industrially relevant programming language.
2. Implement a wide range of intermediate data structures and algorithms to act as modules of larger programs.
3. Use an appropriate integrated development environment to create robust applications.
4. Demonstrate sound programming and software engineering practices independent of the environment or tools used.
5. Explain the theoretical issues surrounding programming language design and development.

## Resources

### Software

This paper will be taught using **Microsoft Visual Studio 2019**. An installer for **Microsoft Visual Studio 2019** is available. See <https://visualstudio.microsoft.com/vs>. Please refer any problems with downloads or installers to **Rob Broadley** in **D205a**.

### Readings

There is no textbook for the course.

## Provisional Schedule

Week	Date	Session 1	Session 2
1	17-02-2020	C++/CLI	Pointers
2	24-02-2020	File Structure	Event Loop
3	02-03-2020	Linked List	
4	09-03-2020	2D Animation Algorithms & Double Buffering	Directional Sprite
5	16-03-2020	Bounds Action	Tile Map
6	23-03-2020	Manual Scrolling Tile Map	Mid Semester Break
Mid Semester Break			
7	13-04-2020	Sprite & NPC Scrolling Tile Map	NPC Collision
8	20-04-2020	Terrain Collision	Finite State Machine & Inheritance
9	27-04-2020	Python 1: Basic Data Structures & Comprehensions	
10	04-05-2020	Python 2: Functional Programming & Other In-Built Functions	
11	11-05-2020	Assessment Work: Roguelike	
12	18-05-2020	Python 3: Exceptions & Automation Testing	Python 4: SQLAlchemy
13	25-05-2020	Python 5: Concurrency & Parallelism	
14	01-06-2020	Assessment Work: Language Exploration	
15	08-06-2020	Assessment Work: Language Exploration	
16	15-06-2020	Theory Exam Preparation	Theory Exam

## Assessments

Assessment	Weight	Due Date	Learning Outcomes
In Class Checkpoints	15%	12-06-2020	1, 2, 3
Rogue-like	45%	15-05-2020	1, 2, 3
Programming Language Exploration	25%	12-06-2020	1, 2, 3
Theory Exam	15%	19-06-2020	3, 4, 5

## Course Requirements and Expectations

### Learning Hours

This course requires 150 hours of learning. This time includes 64 hours of timetabled class time, and 86 hours of self-directed reading, preparation and completion of assessment work.

### Criteria for Passing

To pass this paper, you must achieve an overall average of 50%. There must be a genuine attempt at all assessments. There are no resits.

### Attendance

- Students are expected to attend all classes, both lectures and labs.
- If you miss a class, you will need to get notes from another student.
- If you cannot attend for a few days for any reason, please contact your lecturer.
- You must turn up ready for assessments on the due date and at the correct time. No extra time will be scheduled. If you do not turn up, you have failed the assessment.

### Communication

Microsoft Outlook and Teams are the official communication channels. It is your responsibility to regularly check Microsoft Outlook/Teams and [GitHub](#) for important course-related material, including changes to class scheduling or assessment details. Not checking will not be accepted as an excuse.

### Snow Days/Polytechnic Closure

In the event the Polytechnic is closed or has a delayed opening because of snow or bad weather, you should not attempt to attend class if it is unsafe to do so. It is possible that your lecturer will not be able to attend either, so classes will not physically be meeting. However, this does not become a holiday. Rather, the material will be made available on [GitHub](#) for classes affected by the closure. You are responsible for any material presented in this manner. Information about closure will be posted on the Otago Polytechnic Facebook page <https://www.facebook.com/OtagoPoly>.

### Group Work and Originality

Students in the Bachelor of Information Technology degree are expected to hand in original work. Students are encouraged to discuss assessments with their fellow students, however, all assessments are to be completed as individual work unless group work is explicitly required (i.e. if it doesn't say it is group work then it is not group work – even if a group consultation was involved). Failure to submit your own original work will be treated as plagiarism.

## **Referencing**

Appropriate referencing is required for all work. Referencing standards will be specified by your lecturer.

## **Plagiarism**

Plagiarism is submitting someone else's work as your own. Plagiarism offences are taken seriously and an assessment that has been plagiarised may be awarded a zero mark. A definition of plagiarism is in the Student Handbook, available online or at the school office.

## **Submission Requirements**

All assessments are to be submitted by the time, date, and method given when the assessment is issued. Failure to meet all requirements may result in a penalty of up to 10% per day (including weekends).

## **Extensions**

Extensions are only available for unusual circumstances. These must be applied for, and approved, before the submission deadline.

## **Impairment**

In case of sickness contact your lecturer or BIT Team Leader (Michael Holtz) as soon as possible, preferably before the assessment or exam is due. The policy regarding the granting of a mark that considers impaired performance requires a medical certificate and a medical practitioner's signature on a form. You may refer to the guide on impaired performance on the student handbook.

## **Appeals**

If you are concerned about any aspect of your assessment, please approach the lecturer in the first instance. We support an open-door policy and aim to resolve issues promptly. Further support is available from the BIT Team Leader (Michael Holtz) and Head of College (Richard Nyhof). Otago Polytechnic has a formal process for academic appeals if necessary.

## **Other Documents**

Regulatory documents relating to this course can be found on the Polytechnic website.