

CS1010X: Programming Methodology I

Chapter 0: Course Administration

Lim Dillion
dillionlim@u.nus.edu

2025/26 Special Term 2
National University of Singapore

0	Course Administration	2
0.1	Introduction	2
0.1.1	Overview	2
0.1.2	Resources	2
0.2	Course Schedule	3
0.2.1	Weekly Schedule	3
0.2.2	Exam Schedule	3
0.3	Course Grading	3
0.4	Python Installation Guide	4
0.4.1	Windows	4
0.4.2	macOS	4
0.4.3	Linux	5

Chapter 0: Course Administration

0.1 Introduction

0.1.1 Overview

This course introduces the fundamental concepts of problem-solving by computing and programming using an imperative programming language. It is the definitive introductory course to computing, equivalent to **CS1010**, **CS1010S**, and **CS1010E** Programming Methodology.

The course is taught primarily in **Python** (with a transition to **Java** towards the end).

Course Pillars

The main focus of CS1010 level courses is **correctness**. Topics include:

- Problem formulation and algorithmic thinking.
- Program development: coding, testing, and debugging.
- Fundamental constructs: variables, types, control, and functions.
- Data structures: arrays, strings, and structures.
- Recursion and file processing.

The focus for CS1010 level courses is **correctness**.

0.1.2 Resources

While there is no mandatory textbook, the course follows the philosophy of *Structure and Interpretation of Computer Programs (SICP)*.

SICP (Original)

Structure and Interpretation of Computer Programs. Written in **Scheme**. While the language differs, the conceptual depth is identical.

SICP in Python

SICP in Python. An adaptation of the classic text for **Python**.

Essential Tools:

- **Coursemology**: Hub for missions, XP, contests, and forums.
- **Canvas**: Zoom recordings, grade releases, and feedback.
- **Python Tutor**: Visualization tool for tracing code execution.

0.2 Course Schedule

0.2.1 Weekly Schedule

Online Recitation & Tutorial (Fridays/Saturdays)

Start Date: 16 January 2026

- Review Session (7:00pm): Review of weekly lectures and Q&A.
- Core Session (8:00pm – 9:30pm): Alternating recitation and tutorials (Compulsory).

In-class Recitation (Fridays/Saturdays)

Start Date: 10 April (Friday) / 11 April (Saturday) | Venue: TBC

- **Fri:** 7:30pm – 9:00pm
- **Sat:** 6:00pm – 7:30pm **OR** 7:30pm – 9:00pm

In-class Tutorial (Fridays/Saturdays)

Start Date: 17 April (Friday) / 18 April (Saturday) | Venue: TBC

- **Fri:** 2 sessions, timings TBC
- **Sat:** 4 sessions, timings TBC

0.2.2 Exam Schedule

Venue: School of Computing, Exact Venues TBC

- **Midterm Examination:** 28 March 2026 (Saturday), 10:00am – 12:00pm
 - **Practical Examination:** 29 May 2026 (Friday), 10:00am – 12:00pm
 - **Final Examination:** 19 June 2026 (Friday), 9:00am – 11:00pm
- [**Important:** For finals, check details, such as seating plan and location in EduRec from Office of the University Registrar (OUR).]

0.3 Course Grading

The assessment structure is designed to reward consistent effort.

Table 0.1: Course Grading

Component	Weightage
Coursemology Levels Earn EXP via missions. Reaching Level 35 (TBC) guarantees full marks.	33%
Tutorial Participation Based on attendance during in-person classes.	2%
Midterm Examination	16%
Practical Examination	16%
Final Examination	33%
Total	100%

0.4 Python Installation Guide

You are required to install **Python 3.12** or newer.

0.4.1 Windows

1. Visit <https://www.python.org/downloads/windows/> and download the latest Python 3 installer.
2. **Run the installer.**

CRITICAL STEP

At the bottom of the installation window, you **MUST** check the box that says:
☐ **Add Python to PATH**
If you miss this, you will not be able to run Python from your terminal.

3. Click **Install Now**.
4. Verify installation by opening Command Prompt ('cmd') and running:

```
1 python --version
2 # Output should be Python 3.12.x
```

0.4.2 macOS

1. Visit <https://www.python.org/downloads/mac-osx/> and download the **macOS 64-bit universal2 installer**.
2. Run the .pkg file and follow the prompts.
3. Verify via Terminal:

```
1 python3 --version
```

0.4.3 Linux

Option A: Standard (apt)

For Ubuntu 24.04+, Python 3.12 is likely pre-installed.

```
1 sudo apt update
2 sudo apt install python3 python3-pip
3 python3 --version
```

Option B: Modern Manager (uv)

We recommend **uv** for managing Python versions cleanly.

1. Install uv:

```
1 curl -LsSf https://astral.sh/uv/install.sh | sh
```

2. Restart your terminal (close and reopen).

3. Install & Setup:

```
1 # Install Python 3.12
2 uv python install 3.12
3
4 # Create a virtual environment
5 uv venv .venv --python 3.12
6 source .venv/bin/activate
7
8 # Verify
9 python --version
```