Course Code	Course Title
COMP 1021	**Introduction to Computer Science**

Course Description

This course introduces students to the world of computer science. Students will experience a range of fun and interesting areas from the world of computing, such as game programming, web programming, user interface design and computer graphics. These will be explored largely by programming in the Python language. Exclusion(s): COMP 1022P, COMP 1022Q (prior to 2020-21), COMP 2011, COMP 2012H

List of Topics

Introduction to Python
Introduction to Turtle Graphics
Comments and Text
Making Decisions
Loops
Lists and Tuples
Slicing
Functions
Data Types
File Handling
Dictionaries
State Diagrams
Turtle Object Creation
Event Handling
Stacks
Advanced Operators
Objects
Recursion

Textbooks

- Interactive Python Programming for Beginners
- Written by Gibson Lam and David Rossiter

Reference books

None required.

Grading Scheme

- Labs
  - 3 lab projects will be handed in for marking 3 x 12%/14% = **36%/ 42%**
- Midterm exam **24%/ 0%**

- Final exam **40%/ 58%**

- The first way we assess you is this *(more midterm %)*:
  Midterm 24%, **Lab projects 36%**, **Final exam 40%**
- The second way we assess you is this *(less midterm %)*:
  Midterm 0%, **Lab projects 42%**, **Final exam 58%**
- We will automatically choose the highest mark of these two calculations

Course Intended Learning Outcomes

- On successful completion of this course, students are expected to be able to:

1. Demonstrate programming skills, with an emphasis on the Python programming language
2. Write programs in interesting areas such as game programming, computer graphics and user interface design

Assessment Rubrics

Level of Achievement	Learning Outcome 1	Learning Outcome 2
Exemplary	The student is able to proficiently program in the Python language with no or very few errors, and is able to detect and fix bugs in the code on his/her own. The program can return the correct output on all input instances.	Given the application requirement for a specific area, the student is able to proficiently write a program that meets the requirement.
Competent	The student is able to program in the Python language satisfactorily. The code may contain a few bugs, but the student is able to identify the majority of the bugs on his/her own. Given some further help, the student is able to identify and fix all the bugs. The program can return the correct output on most input instances, but may miss some special or boundary cases.	Given the application requirement for a specific area, the student is able to write a Python program that meets the requirement satisfactorily. The program may crash or behave incorrectly under some harsh tests.
Needs work	The student is able to barely program in the	Given the application requirement for a specific

	Python language on his/her own, but with some help, he/she is able to write a program that behaves correctly on typical input instances. The code often contains bugs and the student is hardly able to identify or correct them.	area, the student is barely able to write a program that meets the requirement.
Unsatisfactory	The student is not able to program in the Python language.	Given the application requirement for a specific area, the student is not able to write a program to achieve the requirement.