

# SEE1002: INTRODUCTION TO COMPUTING FOR ENERGY AND ENVIRONMENT

---

## Effective Term

Semester A 2024/25

## Part I Course Overview

### Course Title

Introduction to Computing for Energy and Environment

### Subject Code

SEE - School of Energy and Environment

### Course Number

1002

### Academic Unit

School of Energy and Environment (E2)

### College/School

School of Energy and Environment (E2)

### Course Duration

One Semester

### Credit Units

3

### Level

B1, B2, B3, B4 - Bachelor's Degree

### Medium of Instruction

English

### Medium of Assessment

English

### Prerequisites

Nil

### Precursors

Nil

### Equivalent Courses

Nil

### Exclusive Courses

Nil

## Part II Course Details

### Abstract

Computing is ubiquitous in Energy and Environment. Whether one works in industry, government or academia, numerical simulations need to be performed, equations solved or data analysed. As part of a modern engineering education, it is therefore highly beneficial for students to acquire familiarity with basic computing practices.

This course will introduce students to computing and programming through the use of an interpreted language, Python. Other software applications (e.g. Flowgorithm and Microsoft Excel) will be covered briefly as well.

### Course Intended Learning Outcomes (CILOs)

| CILOs |                                                                 | Weighting (if app.) | DEC-A1 | DEC-A2 | DEC-A3 |
|-------|-----------------------------------------------------------------|---------------------|--------|--------|--------|
| 1     | Explain the applications of computing in Energy and Environment | 5                   | x      |        |        |
| 2     | Explain the structure of a computer program                     | 15                  |        | x      |        |
| 3     | Write, test and debug simple computer programs                  | 50                  |        | x      |        |
| 4     | Solve elementary engineering problems using simple programs     | 15                  |        | x      |        |
| 5     | Read, write and analyse moderately large datasets               | 15                  | x      | x      |        |

#### A1: Attitude

Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

#### A2: Ability

Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

#### A3: Accomplishments

Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

### Learning and Teaching Activities (LTAs)

| LTAs |  | Brief Description                                                                                                                                                      | CILO No.      | Hours/week (if applicable)  |
|------|--|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|-----------------------------|
| 1    |  | Students will learn on (1) concept of computing; (2) elements of Python programming; (3) basic structure of Python programming; (4) Python for science and engineering | 1, 2, 3, 4, 5 | 21 (for the whole semester) |
| 2    |  | Student will practice Python through computer labs on (1) basic structure of Python programming; (2) Python for science and engineering                                | 2, 3, 4, 5    | 18 (for the whole semester) |

### Additional Information for LTAs

Lectures will cover background material on computing and introduce students to key concepts in the Python language. This material will be applied in the computer labs.

#### Assessment Tasks / Activities (ATs)

| ATs |                                                                                                                                                                 | CILO No.   | Weighting (%) | Remarks (e.g. Parameter for GenAI use) |
|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|---------------|----------------------------------------|
| 1   | Quizzes<br>Students will be tested on their understanding of basic Python concepts/syntax and ability to write short programs.                                  | 2, 3, 4, 5 | 25            | Three quizzes per semester             |
| 2   | Problem sets<br>Longer and more difficult questions based on the computer labs                                                                                  | 2, 3, 4, 5 | 10            | Each practical lab has task weekly     |
| 3   | Lecture participation<br>Students will be assessed on their attendance and participation in class activities (e.g. doing assigned exercises, asking questions). | 2, 3, 4, 5 | 5             |                                        |
| 4   | Midterm                                                                                                                                                         | 1, 2, 3    | 20            |                                        |

#### Continuous Assessment (%)

60

#### Examination (%)

40

#### Examination Duration (Hours)

2

#### Additional Information for ATs

Final exam will emphasize students' ability to solve simple engineering problems and analyse data.

There will be roughly one assignment for each of CILOs 2 - 5.

Examination duration: 2 hrs

Percentage of continuous assessment, examination, etc.: 60% by continuous assessment; 40% by exam

To pass a course, a student must do ALL of the following:

- obtain at least 30% of the total marks allocated towards continuous assessment (combination of assignments, pop quizzes, term paper, lab reports and/ or quiz, if applicable);
- obtain at least 30% of the total marks allocated towards final examination (if applicable); and
- meet the criteria listed in the section on Assessment Rubrics.

#### Assessment Rubrics (AR)

##### Assessment Task

1. Quizzes

##### Criterion

Basic Python concepts and programming

**Excellent (A+, A, A-)**

No difficulty in identifying syntax errors. Programs conform to standard Python style and give the correct output.

**Good (B+, B, B-)**

Minor problems with syntax. Programs are structured correctly but some of the output is incorrect.

**Fair (C+, C, C-)**

Moderate problems with syntax. Programs are structured incorrectly and the output is largely incorrect.

**Marginal (D)**

Numerous problems with syntax. Programs are somewhat relevant but do not solve the problem.

**Failure (F)**

Little understanding of Python syntax. Programs are unrelated to the problem.

---

**Assessment Task**

2. Problem sets

**Criterion**

Solve problems using Python programs

**Excellent (A+, A, A-)**

Programs conform to standard Python style and give the correct output.

**Good (B+, B, B-)**

Programs are structured correctly but some of the output is incorrect.

**Fair (C+, C, C-)**

Programs are structured incorrectly and the output is largely incorrect.

**Marginal (D)**

Programs are somewhat relevant but do not solve the problem.

**Failure (F)**

Programs are unrelated to the problem.

---

**Assessment Task**

3. Lab participation

**Criterion**

Attendance and participation in activities

**Excellent (A+, A, A-)**

Attendance of all lab sessions; works on lab exercises and asks questions.

**Good (B+, B, B-)**

Attendance of most lab sessions; usually works on lab exercises.

**Fair (C+, C, C-)**

Attendance of some lab sessions; sometimes works on lab exercises

**Marginal (D)**

Occasionally attends lab sessions; occasionally works on lab exercises

**Failure (F)**

Rarely attends labs or works on lab exercises

---

**Assessment Task**

4. Midterm

**Criterion**

Basic Python concepts and programming

**Excellent (A+, A, A-)**

No difficulty in identifying syntax errors. Programs conform to standard Python style and give the correct output.

**Good (B+, B, B-)**

Minor problems with syntax. Programs are structured correctly but some of the output is incorrect.

**Fair (C+, C, C-)**

Moderate problems with syntax. Programs are structured incorrectly and the output is largely incorrect.

**Marginal (D)**

Numerous problems with syntax. Programs are somewhat relevant but do not solve the problem.

**Failure (F)**

Little understanding of Python syntax. Programs are unrelated to the problem.

---

**Assessment Task**

5. Final exam

**Criterion**

Solve elementary engineering problems and analyse data using Python

**Excellent (A+, A, A-)**

No difficulty in identifying syntax errors. Programs conform to standard Python style and give the correct output.

**Good (B+, B, B-)**

Minor problems with syntax. Programs are structured correctly but some of the output is incorrect.

**Fair (C+, C, C-)**

Moderate problems with syntax. Programs are structured incorrectly and the output is largely incorrect.

**Marginal (D)**

Numerous problems with syntax. Programs are somewhat relevant but do not solve the problem.

**Failure (F)**

Little understanding of Python syntax. Programs are unrelated to the problem.

---

## Part III Other Information

### Keyword Syllabus

CILO1: operating system, computer language, interpreter, CPU, floating point arithmetic, numerical simulation, visualisation, data analysis, parallel computing

CILO2: structured programming, function, module, variables, comments

CILO3: debugging

CILO4: NumPy, array, spreadsheet

CILO5: text and binary files, sequential access

## Reading List

### Compulsory Readings

| Title |     |
|-------|-----|
| 1     | Nil |

### Additional Readings

| Title |                                                                                                                                                                                                                                                                                                                                                                                        |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1     | The Python Tutorial, <a href="https://docs.python.org/3/tutorial/index.html">https://docs.python.org/3/tutorial/index.html</a>                                                                                                                                                                                                                                                         |
| 2     | Kent D. Lee, Python Programming Fundamentals, Springer, 2015. Available from <a href="https://julac.hosted.exlibrisgroup.com/permalink/f/10vp6a/TN_cdi_askewsholts_vlebooks_9781447166429">https://julac.hosted.exlibrisgroup.com/permalink/f/10vp6a/TN_cdi_askewsholts_vlebooks_9781447166429</a>                                                                                     |
| 3     | John V. Guttag, Introduction to computation and programming using Python, 2013. Available from <a href="http://encore.lib.cityu.edu.hk/iii/encore/record/C__Rb4659958__SGuttag%2C%20John.__P0%2C2__Orightresult__X4?lang=eng&amp;suite=pearl">http://encore.lib.cityu.edu.hk/iii/encore/record/C__Rb4659958__SGuttag%2C%20John.__P0%2C2__Orightresult__X4?lang=eng&amp;suite=pearl</a> |