Effective Term
Semester A 2022/23

Part I Course Overview

Course Title
Java Programming and Applications

Subject Code
EE - Electrical Engineering

Course Number
3206

Academic Unit
Electrical Engineering (EE)

College/School
College of Engineering (EG)

Course Duration
One Semester

Credit Units
3

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

Medium of Instruction
English

Medium of Assessment
English

Prerequisites
EE2331 Data Structures and Algorithms
or equivalent

Precursors
Nil

Equivalent Courses
EE2311 Object-oriented Programming and Design

Exclusive Courses
Nil
Part II Course Details

Abstract
The aim of this course is to provide students with an understanding of the object-oriented design and programming techniques. Java, a prime object-oriented programming language, is used to illustrate this programming paradigm.

Course Intended Learning Outcomes (CILOs)

<table>
<thead>
<tr>
<th>CILOs</th>
<th>Weighting (if app.)</th>
<th>DEC-A1</th>
<th>DEC-A2</th>
<th>DEC-A3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Apply object-oriented programming paradigm and common design patterns to system designs.</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>2. Apply structural programming approach and data structures to solve more complex computation problems.</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>3. Explain how and why genericity is implemented in computer programs and recognize common design patterns.</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Apply system-level techniques such as multi-threading, database and Event-driven user interface in software development.</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

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.

Teaching and Learning Activities (TLAs)

<table>
<thead>
<tr>
<th>TLAs</th>
<th>Brief Description</th>
<th>CILO No.</th>
<th>Hours/week (if applicable)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture and Tutorial</td>
<td>Teaching activities are primarily based on lectures followed by simple examples to show students the basic skills. Tutorials are conducted in the laboratory. Students will acquire the programming skills via hands on experiences in writing Java codes.</td>
<td>1, 2, 3, 4</td>
<td>3 hrs Lect/wk 1 hrs Tut/wk</td>
</tr>
</tbody>
</table>
Assignments and mini project

Assignments are to be completed by students individually.

Students may form a small group or individually carry out a mini-project that goes through different phases of software development. Students will have chances to review the design from peers and therefore reinforce their learning.

Extra readings are provided for self-study and reference. The readings are generally related to lectured topics and allow students to pursue more details as well as bridge the conceptual gap between theories and applications.

### Assessment Tasks / Activities (ATs)

<table>
<thead>
<tr>
<th>ATs</th>
<th>CILO No.</th>
<th>Weighting (%)</th>
<th>Remarks (e.g. Parameter for GenAI use)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tests (min 2)</td>
<td>1, 2, 3, 4</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>#Assignments (min.3)</td>
<td>1, 2, 3, 4</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Lab Exercises/Reports</td>
<td>1, 2, 3, 4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Continuous Assessment (%)**

50

**Examination (%)**

50

**Examination Duration (Hours)**

2.5

### Additional Information for ATs

Remark:

To pass the course, students are required to achieve at least 30% in course work and 30% in the examination.

# may include homework, tutorial exercise, project/mini-project, presentation

### Assessment Rubrics (AR)

**Assessment Task**

Examination
Criterion
Achievements in CILOs

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
Coursework

Criterion
Achievements in CILOs

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Part III Other Information

Keyword Syllabus

Object-oriented principles and design
Objects and classes; information hiding; encapsulation; data abstraction; inheritance and polymorphism; discovering class relationships; unified modeling language (UML) and diagrams; design patterns; software development process.

Overview of the Java language
Java technologies and platform; basic Java syntax; classes and methods; String and wrapper classes; class definition and packages; method overloading and overriding; superclasses and subclasses; dynamic binding and generic programming; abstract classes and interfaces; binary and text I/O; exceptions and assertions.

GUI programming
Graphical user interface components; frame and layout management; event-driven programming; applets.
Software design using Java
Java Collection Framework (JCF); applications of standard data structures, e.g. scheduling and optimization problems.
Multi-threading; thread safety and liveness; thread corporation and synchronization; task partitioning and performance optimization. Data processing applications; Java Database Connectivity (JDBC); accessing and updating relational database.

Reading List

Compulsory Readings

<table>
<thead>
<tr>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture notes</td>
</tr>
</tbody>
</table>

Additional Readings

<table>
<thead>
<tr>
<th>Title</th>
</tr>
</thead>
</table>