Title (Units): **COMP3046 Advanced Programming for Software Development (3,1,3)**

Course Aims: This course aims to further develop students' knowledge and skills in programming for software development. Advanced topics in programming will be introduced and studio-based learning approach which offers high degree of interaction, collaboration and constant feedbacks to students will be adopted.

Technically, students learn about advanced programming topics. Students can work in teams and will be evaluated by project-based assessment and programming assignments. At the end of the study of this course, the students should:

i) acquire knowledge in advanced topics in programming
ii) be able to apply the knowledge and skills in programming in the development of advanced software applications
iii) be able to work in teams to build advanced software applications

Prerequisite: COMP2045 Programming and Problem Solving AND
COMP2046 Problem Solving Using Object Oriented Approach

Course Intended Learning Outcomes (CILOs):
Upon successful completion of this course, students should be able to:

No.	Course Intended Learning Outcomes (CILOs)
	Knowledge
1	Describe and understand the usage of advanced topics in programming in software development
2	Describe the elements and principles of parallel programming
3	Describe and understand the characteristics in different programming languages and select the appropriate one for a given task
	Skill
4	Design and develop advanced software applications in teams which follow the development processes standards
5	Formulate problems as steps so as to be solved systematically
6	Design and develop applications using parallel computing to achieve high-performance computing

Calendar Description: This course aims to further development students' skills in programming for software development by introducing advanced topics in programming. In addition, students' performance will be evaluated by group-project-based software application development to allow students to gain hands-on experience in working in teams. This course adopts studio-based learning approach which offers high degree of interaction, collaboration and constant feedbacks to students.

Teaching and Learning Activities (TLAs):

CILOs	Type of TLA
1 - 3	Students will attend lectures to learn the concepts of advanced topics in different programming languages.
1 - 6	Students will work on a group project, and each project will be discussed in the design crits on three progressive stages: the pitch, the studio and the presentation, under studio-based pedagogy.
4 - 6	Students will attend programming sessions to gain practical skills on software application development.

Assessment:

No.	Assessment Methods	Weighting	CILOs to be addressed	Description of Assessment Tasks
1	Advanced software	60%	1 - 6	Group projects to evaluate students' creativity and practical skill of software application development.

	application design and development projects			
2	Programming assignments	40%	4 - 6	Basic hands-on programming assignments.

Assessment Rubrics:

Excellent (A)	• - The student acquires excellent knowledge in the advance topics in programming • - Able to design and construct a new software application • - Demonstrate an excellent capability in developing software with other teams • - Fully engaged in the design crit sessions
Good (B)	• - The student acquires sufficient knowledge in the advance topics in programming • - Able to design and construct a new software application by combining and extending examples • - Demonstrate a good capability in developing software with other teams • - Full mastery of all basic software development skills
Average (C)	• - The student acquires average knowledge in the advance topics in programming • - Able to construct a new software application with substantial help and guidance • - Able to work with other teams in developing software • - Adequate knowledge on software development skills
Satisfactory (D)	• - The student acquires some knowledge in the advance topics in programming • - Produce a less than workable software application • - Able to work with other teams in developing software with substantial help and guidance • - Demonstrate a satisfactory understanding on software development skills
Unsatisfactory(F)	• - The student acquires some knowledge in the advance topics in programming but below the basic minimum level • - Unable to create a software application • - Unable to work with other teams in developing • - Unable to identify and explain the basic skills in software development

Course Content and CILOs Mapping:

Content		**CILO No.**
I	Advanced topics in programming	1,4,5
II	Topics in Parallel Programming	2,4-6
III	Topics in Selected Programming Languages	3

References:

- C. S. Horstmann and G. Cornell, Core Java 2 (Volume I-Fundamentals), Prentice Hall, 9th Edition, 2013.
- C. S. Horstmann and G. Cornell, Core Java 2 (Volume II-Advanced features), Prentice Hall, 9th Edition, 2013
- David B. Kirk and Wen-mei W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach (Applications of GPU Computing Series), Morgan Kaufmann, 2nd Edition, 2012
- Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 1st Edition, 2011
- David H. Eberly, GPGPU Programming for Games and Science, A K Peters/CRC Press, 1st Edition, 2014
- Nicholas Wilt, CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison-Wesley Professional, 1st Edition, 2013

Course Content:

Topic

I. Advanced topics in programming
   A. Stream and Files
   B. Generic Programming
   C. Event Handling
   D. Graphical User Interface (GUI)
   E. Multi-threading

II. Topics in Parallel Programming
   A. Parallel Hardware and Parallel Software
   B. Introduction to GPU Programming
   C. General-purpose GPU (GPGPU) Programming
   D. Parallel Program Development with OpenMP

III. Topics in Selected Programming Languages
   A. Introduction - Basics in the Selected Programming Languages
   B. Comparing the Selected Programming Languages