QFO-AP-FI-MO02	اسم النموذج: Course Syllabus	جامعة فيلادلفيا Philadelphia University
رقم الاصدار : 1 (Revision)	الجهة المصدرة: كلية تكنولوجيا المعلومات	
التاريخ :2017/11/05	الجهة المدققة: عمادة التطوير والجودة	
عدد صفحات النموذج:		

Course Title: Concurrent and Distributed Programming	**Code:** 0750413
Course Level: 4	**Prerequisite: 750215**
Lecture Time:	**Credit hours:** 3

Academic Staff Specifics

Name	Rank	Office	Office Hours	E-mail Address

Course module description:

The course will focus on the major problems specific to Concurrent and Distributed programming (synchronization, communication, mutual exclusion), and on the means by which such problems can be avoided or overcome (hardware support, semaphores, monitors…). Language features for Concurrent and Distributed Programming, and a set of selected Concurrent and Distributed algorithms are also presented. Students will be given an experience in concurrent and distributed programming.

Course module objectives:

The aim of this course is to introduce and detail the main concepts and practices in concurrent and distributed programming.

Course/ module components

- **Books (title , author (s), publisher, year of publication)**

Concurrent Programming: Algorithms, Principles, and Foundations 2013th Edition
by Michel Raynal

- **Support material (s):**
  Textbook slides

Teaching methods:

Lectures, tutorials, problem solving, practical works

Learning outcomes

A- Knowledge and understanding

A2. Know & understand a wide range of principles and tools available to the software developer, such as design methodologies, choice of algorithm, language, software libraries and user interface technique:

.A3. Understand the principles of various current applications and research areas of the subject including artificial intelligence, databases, software engineering, networks, and distributed systems

A4. Know & understand a wide range of software and hardware used in development of computer systems

A5. Know & understand the professional and ethical responsibilities of the practising computer professional including understanding the need for quality, security, and computer ethics.

B- *Cognitive skills (thinking and analysis).*

B1. Analyze a wide range of problems and provide solutions through suitable algorithms, structures, diagrams, and other appropriate methods

B3. Identify a range of solutions and critically evaluate and justify proposed design solutions

B4. Practice self learning by using the e-courses

C- *Communication skills (personal and academic).*

C2. Prepare and deliver coherent and structured verbal and written technical reports.

C4. Use the scientific literature effectively and make discriminating use of Web resources

C5. Design, write, and debug computer programs in appropriate languages

D- *Practical and subject specific skills (Transferable Skills).*

D3. Work effectively with and for others.

D4. Strike the balance between self-reliance and seeking help when necessary in new situations

D5. Display personal responsibility by working to multiple deadlines in complex activities

Learning outcomes achievement

- Development: A2, A4, and A5 are developed through the lectures and Practical Works.
  B1, B3, C5, D3, and D4 are developed trough Tutorials and Practical works,
  A3, B4, C2, C4, C5, and D5 are developed through Homework

- Assessment : A2, A4, A5, B1, C5, and D4 and are assessed through Quizzes, written exams, and Practical Works Exams.
  B4, C2, C4, C5, and D5 are assessed through Homework Exam.

.

Assessment instruments

- Quizzes.
- Home works: Practical projects
- Final examination: 50 marks

Allocation of Marks	
Assessment Instruments	**Mark**
First examination	20
Second examination	20
Final examination: 50 marks	40
Quizzes, Home works	20
Total	**100**

Documentation and academic honesty

- Documentation style (with illustrative examples)
- Protection by copyright

- Avoiding plagiarism.

Course/module academic calendar

Week	Basic and support material to be covered	Homework/reports and their due dates
(1)	Introduction: What is concurrent programming? The Concurrent Programming Abstraction (1)	
(2)	The Concurrent Programming Abstraction (2) – Tutorial	
(3)	The Critical Section Problem	
(4)	The Advanced Algorithms for the Critical Section Problem	
(5)	Semaphores (1)	
(6) **First examination**	Semaphores (2) – Tutorial	
(7)	Critical regions Monitors (1)	
(8)	Monitors (2) – Tutorial	
(9)	Channels (1)	
(10)	Channels (2) – Tutorial	
(11) **Second examination**	Spaces	
(12)	Distributed Algorithms (1)	
(13)	Distributed Algorithms (2) – Tutorial	
(14)	Global properties	
(15) **Specimen Examination (Optional)**	Consensus (1)	
(16) **Final Examination**	Consensus (2) – Tutorial	

Expected Workload:

On average students need to spend 2 hours of study and preparation for each 50-minute lecture/tutorial.

Attendance policy:

Absence from lectures and/or tutorials shall not exceed 15%. Students who exceed the 15% limit without a medical or emergency excuse acceptable to and approved by the Dean of the relevant college/faculty shall not be allowed to take the final examination and shall receive a mark of zero for the course. If the excuse is approved by the Dean, the student shall be considered to have withdrawn from the course.

Module references

Books:

Elements of Distributed Algorithms: Modeling and Analysis with Petri Nets by Wolfgang Reisig, 2010
Concurrent Programming: Algorithms, Principles, and Foundations, Michel Raynal, 2012
Principles of Concurrent and Distributed Programming, M. Ben-Ari, Addisson Wesley, Second edition, 2006.

Websites

http://users.ece.utexas.edu/~garg/jbk.html
http://www.gobookee.org/principles-of-concurrent-and-distributed-programming/
http://www3.ntu.edu.sg/home/ehchua/programming/java/J5e_multithreading.html
http://en.wikipedia.org/wiki/Java_concurrency