جامعة فيلادلفيا Philadelphia University	اسم النموذج : خطة المادة الدراسية	رمز النموذج : QFO-AP-VA-008
	الجهة المصدرة: نائب الرئيس للشؤون الأكاديمية	رقم الإصدار: (Rev) 2
	الجهة المدققة : اللجنة العليا لضمان الجودة	تاريخ الإصدار: 2021-5-4
		عدد صفحات النموذج : 4

Course Title: Concurrent and Distributed Programming			**Course code:** 0750413		
Course Level: 4			**Course prerequisite (s) and/or co requisite (s):** (**750215**)		
Lecture Time:			**Credit hours:** 3		
UR ☐	**FR** ☐	**DR** ■	**C** ■	**E** ☐	

Academic Staff Specifics

Name	Rank	Office	Office Hours	E-mail Address
Maram Bani Younes	Associate Professor	7306		**mbaniyounes@philadelphia.edu.jo**

The Learning Style Used in Teaching the Course

The Learning Style			
Blended Learning ☐			
Electronic Learning ☐			
Face-to-Face Learning ■			
Face-to-Face	**Electronic**	**Blended**	**Percentage**

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
A1. Recognize 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:

A2. Recognize the principles of various current applications and research areas of the subject including artificial intelligence, databases, software engineering, networks, and distributed systems

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.

B2. Identify a range of solutions and critically evaluate and justify proposed design solutions

C- Communication skills (personal and academic).
C1. Prepare and deliver coherent and structured verbal and written technical reports.

C2. Design, write, and debug computer programs in appropriate languages

D- Practical and subject specific skills (Transferable Skills).
D1. Display personal responsibility by working to multiple deadlines in complex activities

Learning Outcomes Achievements:

A1, A2, B1, and B2 are achieved through lectures and assessed by quizzes and examinations
C1, C2, and D1 are achieved and assessed by homework: practical and research works

Assessment instruments

- Quizzes.& Home works
- Final examination: 40 marks

Allocation of Marks	
Assessment Instruments	**Mark**
Mid Examination	**30**

Final Examination	**40**
Quizzes, Home works	**30**
Total	**100**

Assignments All assignments will be announced or handed out in class. Many assignments will require programming in Python. All individual assignments, whether programming or not, are to be done individually. While you may discuss the assignment in general terms with others, your solutions should be composed, designed, written and tested by you alone. If you need help, consult the TA or the instructor.

Documentation and academic honesty

Submit your home work covered with a sheet containing your name, number, course title and number, and type and number of the home work (e.g. tutorial, assignment, and project).

Any completed homework must be handed in to my office (room IT 602) by 15:00 on the due date. After the deadline "zero" will be awarded. You must keep a duplicate copy of your work because it may be needed while the original is being marked.

You should hand in with your assignments:

1- A printed listing of your test programs (if any).
2- A brief report to explain your findings.
3- Your solution of questions.

For the research report, you are required to write a report similar to a research paper. It should include:

- **Abstract**: It describes the main synopsis of your paper.
- **Introduction**: It provides background information necessary to understand the research and getting readers interested in your subject. The introduction is where you put your problem in context and is likely where the bulk of your sources will appear.
- **Methods (Algorithms and Implementation)**: Describe your methods here. Summarize the algorithms generally, highlight features relevant to your project, and refer readers to your references for further details.
- **Results and Discussion (Benchmarking and Analysis)**: This section is the most important part of your paper. It is here that you demonstrate the work you have accomplished on this project and explain its significance. The quality of your analysis will impact your final grade more than any other component on the paper. You should therefore plan to spend the bulk of your project time not just gathering data, but determining what it ultimately means and deciding how best to showcase these findings.
- **Conclusion**: The conclusion should give your reader the points to "take home" from your paper. It should state clearly what your results demonstrate about the problem you were tackling in the paper. It should also generalize your findings, putting them into a useful context that can be built upon. All generalizations should be supported by your data, however; the discussion should prove these points, so that when the reader gets to the conclusion, the statements are logical and seem self-evident.
- **Bibliography:** Refer to any reference that you used in your assignment. Citations in the body of the paper should refer to a bibliography at the end of the paper.

- **Protection by Copyright**

1. Coursework, laboratory exercises, reports, and essays submitted for assessment must be your own work, unless in the case of group projects a joint effort is expected and is indicated as such.
2. Use of quotations or data from the work of others is entirely acceptable, and is often very valuable provided that the source of the quotation or data is given. Failure to provide a source or put quotation marks around material that is taken from elsewhere gives the appearance that

the comments are ostensibly your own. When quoting word-for-word from the work of another person quotation marks or indenting (setting the quotation in from the margin) must be used and the source of the quoted material must be acknowledged.

3. Sources of quotations used should be listed in full in a bibliography at the end of your piece of work.

- **Avoiding Plagiarism**.

1. Unacknowledged direct copying from the work of another person, or the close paraphrasing of somebody else's work, is called plagiarism and is a serious offence, equated with cheating in examinations. This applies to copying both from other students' work and from published sources such as books, reports or journal articles.
2. Paraphrasing, when the original statement is still identifiable and has no acknowledgement, is plagiarism. A close paraphrase of another person's work must have an acknowledgement to the source. It is not acceptable for you to put together unacknowledged passages from the same or from different sources linking these together with a few words or sentences of your own and changing a few words from the original text: this is regarded as over-dependence on other sources, which is a form of plagiarism.
3. Direct quotations from an earlier piece of your own work, if not attributed, suggest that your work is original, when in fact it is not. The direct copying of one's own writings qualifies as plagiarism if the fact that the work has been or is to be presented elsewhere is not acknowledged.
4. Plagiarism is a serious offence and will always result in imposition of a penalty. In deciding upon the penalty the Department will take into account factors such as the year of study, the extent and proportion of the work that has been plagiarized, and the apparent intent of the student. The penalties that can be imposed range from a minimum of a zero mark for the work (without allowing resubmission) through caution to disciplinary measures (such as suspension or expulsion).

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 exam.	Semaphores (2) –	Tutorial
(7)	Critical regions Monitors (1)	
(8)	Monitors (2) – Tutorial	
(9)	Channels (1)	
(10)	Channels (2) –	Tutorial
(11) Second exam.	Spaces	
(12)	Distributed Algorithms (1)	
(13)	Distributed Algorithms (2) –	Tutorial
(14)	Global properties	

(15)	Consensus (1)	
(16) Final Exam.	Revision	

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