Philadelphia University		Approval date: 8/10/2024
Faculty of Science	PHILADELPHIA UNIVERSITY THE WAY TO THE FUTURE	Issue:
Department of Math		Credit hours: 3
Academic year 2024/2025	Course Syllabus	Bachelor

Course information

Course#	Course title		Prerequisite
250372	**Computer Aided Mathematics**		**ODEs 250203**
Course type		**Class time**	**Room #**
☐ University Requirement ☒ Major Requirement	☐ Faculty Requirement ☐ Elective ☒ Compulsory	ST 12:40 – 13:30	2827
Degree / NQF Level	☐ Diploma degree (6)	☒ Bachelor degree (7)	

Instructor Information

Name	Office No.	Phone No.	Office Hours	E-mail
Feras Awad	822	2132	SSMT 11:30-12:30	fawad@philadelphia.edu.jo

Course Delivery Method

Course Delivery Method		
☐ **Physical**	☐ **Online**	☒ **Blended**
Learning Model		
Precentage	**Synchronous**	**Asynchronous**	**Physical**
	0%	33%	67%

Course Description

The structure of Mathematica. Mathematica as a Calculator. Variables and functions. Lists. Logic and set theory. Number Theory. Computer algebra and Solving Equations. Single Variable Calculus.

Course Learning Outcomes

Number	Outcomes	Corresponding Program outcomes
Knowledge		
K1	Learn the use of commands and functions for solving and visualizing mathematical problems.	**K_p3**
Skills		
S1	Use Wolfram Language to solve problems graphically, numerically and analytically.	**S_p4**
Competencies		
C1	Work in a team to implement one of the tasks of the course.	**C_p2**

Learning Resources

Course textbook	• Feras Awad (07/07/2024) A Glimpse to Mathematica [Wolfram Language]. Instructor Lectures and Notes.
Supporting References	• Wellin, P. (2013) Programming with Mathematica: An Introduction (1st ed.). Cambridge University Press. • Hastings, C., Mischo, K., Michael M. (2015) Hands-on start to Wolfram Mathematica (1st ed.). Champaign: Wolfram Media, Inc.
Supporting websites	WolframCloud: www.wolframcloud.com/
Teaching Environment	☐Classroom ☒ laboratory ☒Learning platform ☐Other

Meetings and Subjects Timetable

Week	Topic	Learning Methods	Tasks	Learning Material
1	Explanation of the study plan for the course, and what is expected to be accomplished by the students. **Introduction:** What Is the Wolfram Language? Wolfram Cloud. What is Mathematica? The Structure of Mathematica. Common Kinds of Interfaces to Mathematica. Notebook Interfaces. Palettes	Lecture		Course Syllabus Chapter 1
2	**Mathematica as a Calculator:** Commands for Basic Arithmetic. Precedence. Built-in Constants. Built-in Functions. Numerical and Scientific Notations. Prefix and Postfix Forms for Built-in Functions. Mathematica Help	Lecture		Chapter 2
3	**Variables and Functions:** Rules for Names. Immediate Assignment. Functions. Substitution Rule.	Lecture		Chapter 3
4	Anonymous Functions. Functions with Conditions.	Lecture	Quiz 1	Chapter 3
5	**Lists:** What is a List? Functions Producing Lists. Displaying Lists. Working with Elements of a List.	Lecture		Chapter 4
6	Useful Functions. Listable Functions. Nested Loops.	Lecture		Chapter 4
7	Vectors. Matrices. Special Types of Matrices. Basic Matrix Operations	Lecture		Chapter 4
8	**Logic and Set Theory:** Being Logical. Truth Tables. Element "$\in$". Handling Sets. Quantifiers	Lecture	Midterm	Chapter 5
9	**Number Theory** Primes. Integer Factorization. Digits in Numbers. Fibonacci Sequence.	Lecture		Chapter 6
10	Number Theoretic Functions. Selecting from Lists	Lecture		Chapter 6
11	**Computer Algebra and Solving Equations** Working with Polynomials and Powers. Working with Rational Functions. Working with Transcendental Functions.	Lecture	Quiz 2	Chapter 7

12	Equations and Their Solutions. Inequalities	Lecture		Chapter 7
13	**Single Variable Calculus:** Limits. Differentiation. Maximum and Minimum. Integration.	Lecture		Chapter 8
14	Sequences. Series. Taylor Polynomials	Lecture	Quiz 3	Chapter 8
15	Review	Lecture		
16	Final Exam			

* Includes: Lecture, flipped Class, project- based learning, problem solving based learning, collaborative learning

Course Contributing to Learner Skill Development

Using Technology
• Use Wolfram Language to solve mathematical problems.
Communication Skills
• Choose a mathematical problem and present it to the students and explaining its solution method using Wolfram Language.
Application of Concepts Learnt
• Choose a famous math problem on YouTube and solve it using Wolfram Language.

Assessment Methods and Grade Distribution

Assessment Methods	Grade Weight	Assessment Time (Week No.)	Link to Course Outcomes
Mid Term Exam	**30%**	**8**	**K1**
Various Assessments *	**30%**	**Continuous**	**S1, C1**
Final Exam	**40%**	**16**	**K1**
Total	**100%**		

* Includes: quiz, in class and out of class assignment, presentations, reports, videotaped assignment, group or individual projects.

Alignment of Course Outcomes with Learning and Assessment Methods

Number	Learning Outcomes	Learning Method*	Assessment Method**
Knowledge			
K1	Learn the use of commands and functions for solving and visualizing mathematical problems.	Lecture	**Exam**
Skills			
S1	Use Wolfram Language to solve problems graphically, numerically and analytically.	Lecture	**Computer Assignment**
Competencies			
C1	Work in a team to implement one of the tasks of the course.	Project	**Group Project**

* Includes: Lecture, flipped Class, project- based learning, problem solving based learning, collaborative learning

** Includes: quiz, in class and out of class assignment, presentations, reports, videotaped assignment, group or individual projects.

Course Polices

Policy	Policy Requirements
Passing Grade	The minimum passing grade for the course is (50%) and the minimum final mark recorded on transcript is (35%).
Missing Exams	• Missing an exam without a valid excuse will result in a zero grade to be assigned to the exam or assessment.

	• A Student who misses an exam or scheduled assessment, for a legitimate reason, must submit an official written excuse within a week from an exam or assessment due date. • A student who has an excuse for missing a final exam should submit the excuse to the dean within three days of the missed exam date.	
Attendance	The student is not allowed to be absent more than (15%) of the total hours prescribed for the course, which equates to six lectures days (M, W) and six lectures (S, T). If the student misses more than (15%) of the total hours prescribed for the course without a satisfactory excuse accepted by the dean of the faculty, s/he will be prohibited from taking the final exam and the grade in that course is considered (zero), but if the absence is due to illness or a compulsive excuse accepted by the dean of the college, then withdrawal grade will be recorded.	
Academic Honesty	Philadelphia University pays special attention to the issue of academic integrity, and the penalties stipulated in the university's instructions are applied to those who are proven to have committed an act that violates academic integrity, such as: cheating, plagiarism (academic theft), collusion, and violating intellectual property rights.	

Program Learning Outcomes to be Assessed in this Course

Number	Learning Outcome	Course Title	Assessment Method	Target Performance level
S_p4	The use of technology and software in the various fields of mathematics.	Computer Aided Mathematics	Project	100% of the students get 70% or more on the rubric

Description of Program Learning Outcome Assessment Method

Number	Detailed Description of Assessment
S_p4	The student is given a problem, and use Wolfram Language to write a code that solves the problem.

Assessment Rubric of the Program Learning Outcome

	Poor (1 pt.) Student is very confused and does not understand the topic, nor is able to clearly grasp how to apply it or when to use it.	Fair (2 pts) Student has a decent grasp of the process but makes some major mistakes.	Good (3 pts) Student is almost perfect in their understanding of the topic, with some minor confusion or mistakes.
Code Structure Structure of code, use of functions and procedures, code segmentation	Long code segments, improper usage of functions, functions with side effects.	Code structure needs work.	Code structure has perfectly followed guidelines. Short code segments, proper use of functions.
Code Reuse How well code reuse is implemented	Too much redundancy in code	Occasional code redundancy	No code redundancy
Correctness How correct is the output of the program	Program does not work correctly; output is wrong most of the time or there is no output.	Program works correctly in general in most areas but not in all areas.	Program works correctly in all areas and generates correct output.
Execution How smoothly does the program execute - are there any bugs	Program does not execute.	Program executes but crashes in some areas.	Program executes perfectly.