Course Specifications

Course Title:	**Programming for Cyber Security**
Course Code:	**CSEC 427**
Program:	**Information and Computer Sciences**
Department:	**Computer Science and Information**
College:	**College of Science at AzZulfi**
Institution:	**Majmaah University**

Table of Contents

A. Course Identification

1. Credit hours:3
2. Course type **a.** University ☐ College ☐ Department ☐ Others ☐ **b.** Required ☐ Elective ☒
3. Level/year at which this course is offered:
4. Pre-requisites for this course (if any)**:** Object-Oriented Programming – ICS 211
5. Co-requisites for this course (if any)**:**

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**	✔	80
2	**Blended**	✔	5
3	**E-learning**	✔	5
4	**Correspondence**	--	5
5	**Other**	✔	5

7. Actual Learning Hours (based on academic semester)

No	Activity	Learning Hours
Contact Hours		
1	**Lecture**	34
2	**Laboratory/Studio**	10
3	**Project**	16
4	**Others** (specify)	--
	Total	60
Other Learning Hours*		
1	**Study**	45
2	**Assignments**	15
3	**Library**	05
4	**Projects/Research Essays/Theses**	10
5	**Others** (specify)	00
	Total	(60+75 = 135)

***** The length of time that a learner takes to complete learning activities that lead to achievement of course learning outcomes, such as study time, homework assignments, projects, preparing presentations, library times

B. Course Objectives and Learning Outcomes

1. Course Description

This course will provide students with a good understanding of Cyber security programming as a way of writing codes in a software so that it is protected from all kinds of vulnerabilities, attacks or anything that can cause harm to the software, network or the system. Various analysis and design techniques for improving software security, as well as how to use these techniques and tools to improve and verify software designs and security also will be covered in this course.

2. Course Main Objective

1. Introduce the fundamental concepts and techniques in Cyber security programming.
2. Students will gain an in-depth understanding of the computational properties of security issues
3. Student hands on Cyber security issues using python programming language.

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	**Knowledge:**	
1.1	Understand complicated algorithms and functioning of Malware and all other kinds of attacks.	**K3-CS**
2	**Skills :**	
2.1	Python programming.	S1
2.2	To discover cyber-attacks, to articulate the well-known cyber-attack incident and to explain how to mitigate such threats.	S3-CS
3	**Competence:**	
3.1	hands on cyber security tools	C3-CS
3.2	Communicate effectively with others in Computer Science field.	C2

C. Course Content

No	List of Topics	Contact Hours
1	C verses Python with respect to cyber security programming	2
2	**Introduction**: The Programming Cycle for Python , Python IDE, Interacting with Python Programs , Elements of Python, Type Conversion. **Basics**: Expressions, Assignment Statement, Arithmetic Operators, Operator Precedence, Boolean Expression.	4
3	**Conditionals**: Conditional statement in Python (if-else statement, its working and execution), Nested-if statement and Elif statement in Python, Expression Evaluation & Float Representation. **Loops**: Purpose and working of loops , While loop including its working, For Loop , Nested Loops , Break and Continue.	4
4	**Function**: Parts of A Function , Execution of A Function , Keyword and Default Arguments ,Scope Rules. **Strings** : Length of the string and perform Concatenation and Repeat operations in it. Indexing and Slicing of Strings. **Python Data Structure** : Tuples , Unpacking Sequences , Lists , Mutable Sequences , List Comprehension , Sets , Dictionaries	4

	Higher Order Functions: Treat functions as first class Objects , Lambda Expressions	
5	**Sieve of Eratosthenes**: generate prime numbers with the help of an algorithm given by the Greek Mathematician named Eratosthenes, whose algorithm is known as Sieve of Eratosthenes. **File I/O :** File input and output operations in Python Programming **Exceptions and Assertions Modules** : Introduction , Importing Modules , **Abstract Data Types** : Abstract data types and ADT interface in Python. **Classes** : Class definition and other operations in the classes , Special Methods (such as _init_, _str_, comparison methods and Arithmetic methods etc.) , Class Example , Inheritance , Inheritance and OOP.	4
6	**Assembly** Overview of Assembly language, CPU structure, common assembly instructions, control flow, the stack, function calls, and system calls. **Reverse Engineering** Understand complicated algorithms and functioning of Malware. Main tools used in the reverse engineering process, using both static analysis and dynamic analysis methods.	6
7	**Computer Networks** hands on tools (Wireshark as well as Python libraries like scapy or sockets) Five layers model, protocols such as Ethernet, ARP, IP, UDP, TCP, DNS, HTTP and others. **Windows Internals**: Windows Registry, through Win32API, Objects, Memory Management, Processes & Threads, Synchronization & IPC, to Hooking and Injection. Implementing Windows applications in C, and researching the underlying mechanisms of the operating system using WinDBG.	8
	Endpoint Security Overview of the malwares world and APTs. Using both static and dynamic analysis methods learnt in Reverse Engineering, sandboxes, packers and unpacking, try signing malware and deal with obfuscation. **Vulnerabilities, Exploits and Exploit Kits** Vulnerabilities such as buffer overflows exploit kits (Metasploit). mitigation techniques (ASLR etc)	6
	Network Security **Practical Cryptography** Symmetric and Asymmetric encryption, CA and PKI. **Enterprise Security Architecture** Modern enterprise networks, security devices such as VPNs, Firewalls and Intrusion Detection Systems. **Web Application Security** Web attacks, from info gathering and authentication vulnerabilities, through input validation, SQL injection, XSS and more.	6
	Projects hosted by companies A proof of concept for an idea that the company wanted to check, or a small tool the company wants to develop. The projects are diverse and rely on different skills acquired during the course.	16
Total		60

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge**		
1.1	Students will be able to demonstrate knowledge of all issues related to cyber security and tools to sort out or analyze that.	Lectures, Lab demonstrations Case studies Individual presentations	Written Exam Homework assignment class & lab Activity Quizzes
2.0	**Skills**		
2.1	To understand that python and security programming in addition to tools utilized in Cyber security.	Group discussions, Lab demonstrations, Brainstorming Presentations	Home works and assignments
3.0	**Competence**		
3.1	Students will be able to proof an idea that the company wanted to check, or a small tool the company wants to develop	Group discussions, Case Studies, Brainstorming Presentations	project
3.2	Students will function effectively as a member of a team in order to accomplish a common goal		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	First written mid-term exam	6	15%
2	Second written mid-term exam	12	15%
3	Class activities, group discussions, Presentation + lab	Every week	10%
4	Homework + Assignments	After Every chapter	10%
5	Project	Last month	20%
6	Final written exam	16	30%

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :
Office hours: ,
Email: .....@mu.edu.sa

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist‘‘, 2nd edition, Updated for Python 3, Shroff/O‘Reilly Publishers, 2016 (http://greenteapress.com/wp/thinkpython/) 2. William Stallings, Network Security Essentials: Applications and Standards, Prentice Hall, 4th edition, 2010.
Essential References Materials	1. Guido van Rossum and Fred L. Drake Jr, —An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011. 2. Michael T. Goodrich and Roberto Tamassia, Introduction to Computer Security, Addison Wesley, 2011.
Electronic Materials	----------
Other Learning Materials	----------

2.

Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	1. Classrooms with required digital aids and to support traditional method of teaching using blackboard. 2. Classrooms with proper lighting and air conditioning system integrated with the sound System /audio system. 3. Classroom with smart board interface, display screen and a computer to aid the sessions
Technology Resources (AV, data show, Smart Board, software, etc.)	Smart Board with supporting software / computers with updated versions of software as required to understand the subject concepts with quality headphones.
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	Lab and software as requested

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of Teaching	Students Classroom Observation Committee Professional Development Unit External Reviewers – accreditation committee	Formal Classroom Observation - Direct Student Surveys - Indirect

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of Assessment	Curriculum and Test Development Unit Curriculum Committee Assessment Committee External Reviewers	Faculty Feedback - indirect Student Feedback – indirect Course Reports
Extent of Achievement of Course Learning Outcomes	Quality Assurance Unit Curriculum and Test Development Unit	Course Reports Annual Program Review

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	
Reference No.	
Date	