



Course Specification

Course title: Introduction to programming **Code:** CS 400

Program(s) on which the course is given: Biophysics

Major or minor element of program: minor

Department offering the program: Physics

Department offering the course: Mathematics & computer science

Academic year/ Level: Forth year/first Semester

Date of specification approval: 1/2016

1- Basic information			
Academic year: Fourth	Course Title: Introduction to programming		Code: CS 400
Lecture: 2 hours /week	Practical: 2 hour/week	Tutorial: 0	Total: 4 hours/week
2- Professional Information			
1- Overall aims of course	For students undertaking this course, the aims are to: <ul style="list-style-type: none"> - Give the students the programming concepts in a light of learning Bython programming language. - The course includes: introduction to programming, how to think in designing a program, writing a program. - The course includes programming language fundamentals (basic data types – program structure –statements– expressions–I/O operations –Control statements – algorithms). 		
3- Intended learning outcomes of course (ILOs)			
a. Knowledge and Understanding	On completing this course, students will be able to: <ul style="list-style-type: none"> a1 - Know history of programming languages. a2 - Know and Understand fundamentals of programming language: variables, selection statements, loops, functions, arrays. 		
b. Intellectual Skills	On completing this course, students will be able to: <ul style="list-style-type: none"> b1 - Think in building a computer programs. b2 - Build independent blocks or functions that can be used in other programs. 		



c. Professional and Practical Skills	On completing this course, students will be able to: c1 - Deal with compiler of teaching programming language. c2 - Test and debug her programs. c3 - Write a free of errors program code. c4 - Implement and develop simple programs.										
d. General and Transferable Skills	On completing this course, students will be able to: d1 - Write and execute simple program codes d2 - Deal with structured programming.										
4. Contents											
<ol style="list-style-type: none"> 1. Introduction to Software development life cycle and the definition of it. Explaining of Stages of a typical Software development life. 2. Introducing Software development life cycle Models: Waterfall Model, Iterative Model, Spiral Model and V-Model. 3. Principles of Programming: Introduction, Problem Solving, Algorithm, Examples of Algorithm, Types of Algorithm, Properties of Algorithm. 4. Flow Chart: Flow Chart Symbols, General Rules for flowcharting, Some Flowchart Examples, Advantages of Using Flowchart, Limitations of Using Flowchart 5. Learning Programming with Python : Writing a Python Program 6. A Longer Python program: Values and Variables, Variables and Assignment, Identifiers , Floating-point Types, Control Codes within Strings, Arithmetic binary operators, User Input, The eval Function, Comments, Syntax Errors, Arithmetic Examples , More Arithmetic Operators 7. Conditional Execution : Boolean Expressions , Simple if statement , if/else Statement , Nested Conditionals, Multi-way Decision Statements , If statement examples 8. Iteration: While statement , While examples ,For statement , for statement examples 											
5- Teaching and learning methods	<ol style="list-style-type: none"> 1- Lectures. 2- Discussion sessions. 3- Lab sessions. 										
6- Student assessment											
a- Methods	<table> <tr> <td>1- Oral Exam</td> <td>to assess</td> <td>a1- a2, b1- b2, d1- d2</td> </tr> <tr> <td>2- Final Exam</td> <td>to assess</td> <td>a1- a2, b1- b2, c1- c4</td> </tr> <tr> <td>3- Mid-Term Exam</td> <td>to assess</td> <td>a1- a2, b1- b2, c1- c4</td> </tr> </table>	1- Oral Exam	to assess	a1- a2, b1- b2, d1- d2	2- Final Exam	to assess	a1- a2, b1- b2, c1- c4	3- Mid-Term Exam	to assess	a1- a2, b1- b2, c1- c4	
1- Oral Exam	to assess	a1- a2, b1- b2, d1- d2									
2- Final Exam	to assess	a1- a2, b1- b2, c1- c4									
3- Mid-Term Exam	to assess	a1- a2, b1- b2, c1- c4									
b-Assessment schedule	<table> <tr> <td>1- Oral Exam</td> <td>week 16</td> </tr> <tr> <td>2- Final Exam</td> <td>week 16</td> </tr> <tr> <td>3- Mid-Term Exam</td> <td>week 7</td> </tr> </table>	1- Oral Exam	week 16	2- Final Exam	week 16	3- Mid-Term Exam	week 7				
1- Oral Exam	week 16										
2- Final Exam	week 16										
3- Mid-Term Exam	week 7										
c-Weighting of assessments	<table> <tr> <td>- Mid-Term Examination</td> <td>14</td> </tr> <tr> <td>- Final-Term Examination</td> <td>50</td> </tr> <tr> <td>- Oral Examination</td> <td>6</td> </tr> <tr> <td>- Practical Examination</td> <td>30</td> </tr> <tr> <td colspan="2" style="text-align: right;">Total 100%</td> </tr> </table>	- Mid-Term Examination	14	- Final-Term Examination	50	- Oral Examination	6	- Practical Examination	30	Total 100%	
- Mid-Term Examination	14										
- Final-Term Examination	50										
- Oral Examination	6										
- Practical Examination	30										
Total 100%											



7- List of references	
a-Course Note	
b- Recommended Text Book	1- " LEARNING TO PROGRAM WITH PYTHON" Richard L. Halterman
c- Additional References	
d. Periodical journals, Web sites, etc.	1. http://agridr.in/tnauEAgri/eagri50/STAM102/pdf/lec14.pdf 2. file:///C:/Users/TEMP/Downloads/2.%20PROGRAMMING%20CONCEPTS.pdf



Knowledge and Skills Matrix That Targeted Course

Content	Week	Knowledge & Understanding	Intellectual Skills	Professional & Practical Skills	General skills
Introduction to Software development life cycle and the definition of it. Explaining of Stages of a typical Software development life	1	a1	b1	c1	d1
Introducing Software development life cycle Models: Waterfall Model, Iterative Model, Spiral Model and V-Model	2	a2	b1	c2	d1
Principles of Programming: Introduction, Problem Solving, Algorithm, Examples of Algorithm, Types of Algorithm, Properties of Algorithm.	3	a2	b1	c3	d2
Flow Chart: Flow Chart Symbols, General Rules for flowcharting, Some Flowchart Examples, Advantages of Using Flowchart, Limitations of Using Flowchart	4	a2	b1	c3	d2
Learning Programming with Python : Writing a Python Program, A Longer Python program: Values and Variables, Variables and Assignment, Identifiers .	5	a2	b1	c3	d2
Floating-point Types, Control Codes within Strings, Arithmetic binary operators, User Input, The eval Function, Comments, Syntax Errors. Arithmetic Examples , More Arithmetic Operators	6	a2	b1	c4	d2
Conditional Execution : Boolean Expressions , Simple if statement , if/else Statement , Nested Conditionals, Multi-way Decision Statements , If statement examples	7	a2	b1	c4	d2
Iteration: While statement, While examples, For statement , for statement examples	8	a2	b1, b2	c4	d2

Knowledge & understanding		Intellectual skills		Professional practical skills		General & transferable skills	
ILOs of course	ILOs of Program	ILOs of course	ILOs of Program	ILOs of course	ILOs of Program	ILOs of course	ILOs of Program
a1	A15	b1	B1	c1	C1	d1	D1
a2	A15	b2	B5	c2	C6	d2	D2
				c3	C6		
				c4	C6		

Al-Azhar University (Girls branch)

Faculty of Science

Department: Mathematics



جامعة الأزهر (فرع البنات)

كلية العلوم

قسم: الرياضيات

Course Coordinator: Name: Dr Lamiaa Mohamed El Bakrawy

Dr Wafaa Ghonim

Dr Enas El-Sharawy

Head of Department of Name: Prof. Dr/ Hode Abeldaim