Course Definition File

Introduction to Programing

IT Specialty (C#)

Syrian Arab Republic		الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليـم العـالـــــي
Syrian Virtual University		الجامعة الافتراضية السورية

1. Basic Information:

Course Name	Introduction to Programing
Course Code	IPG101
Number of Presentational Sessions*	2×8
Number of Synchronous Sessions**	8
Number of Shorter Tests***	2
Number of Exams***	1
Theoretical Sessions Work Load (hrs.)	48
Practical Sessions Work Load (hrs.)	24
Credit Hours	4

***Each presentational session comprises both recorded lecture (1.5 hrs.) and interactive learning content (1.5 hrs.).**

****Each synchronous session comprises the interactive lecture carried out in real time in a virtual class (1.5 hrs.).**

*****Each shorter test is 0.5 hr. long. The final exam is 2 hrs. long.**

N.B.

Generally, each chapter requires two presentational sessions: one for the recorded content and one for the interactive content (unless the chapter is too long, in which case it may require more sessions). This note applies to synchronous sessions as well, where each chapter requires one synchronous session generally.

2. Pre-Requisites:

Course	ID
–	–

3. Course Objectives:

"Programming 1" course aims to acquaint the student with the basic concepts of programming, problems algorithms and their SPL programing, up to micro programing. It enables the student namely to:

1. Acquaint with the concepts of computer programing, algorithms, operating systems, compilers, coding, programing languages, Dot Net, the basics of C#, the main programing instructions such as read, write… up to C# structure, partial programs, and the development of medium size C# application program in Visual Studio Dot Net environment.
2. Be trained how to use different basic and applications algorithms and programing with C#, by means of solved and unsolved examples and problems.

4. Learning Outcomes:

By the end of this course the learner is expected to:

- Be acquainted with the computer as a machine, its hardware and software, their classification, coding, programing concepts, programing languages, compilers, methodical development, solutions strategies…
- Understand the Microsoft Dot Net concept, Dot Net Framework, the C# novice level and master the analysis of programs and their design, the different C# components and priorities…
- Master the general rules of C# instructions, the variable range and different instructions including the 5 basic algorithmic ones.
- Master the applications of control instructions, repeat and continue instructions in C and C#, structured programing instructions and parallel instructions.
- Master the usage of different composite data and character strings, the tables and matrices up to multidimensional table definition and matrix cells…
- Master C# program listing, functions and procedures, their definitions in C#, calling and pass transactions and recovery…
- Be able to solve successfully a set of relevant exercises and problems.

5. Assessment Results:

Chapter Number	Chapter Title	General Objectives	Assessment Type				
			Interactive Content & Recorded Sessions	Applied Activities (Synch. Sessions)	Final Exam*/ Shorter Tests**	Presentations And Interviews***	Reports ***
CH1	Computer Program	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓
CH2	C# Basics	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓
CH3	C# Instructions	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓
CH4	Algorithmic Language	Comprehension –Analytical Thinking – Tools And Application	✓	✓	✓	✓	✓

		Hands– On					
CH5	Structures of Composite Data	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓
CH6	Introduction to Functions and Procedures	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓
CH7	Exercises and Problems	Comprehension –Analytical Thinking – Tools And Application Hands– On	✓	✓	✓	✓	✓

***The final exam is two hours long and is given at the end of the course.**

****Shorter tests are about 30 minutes long and are given after three or four lectures throughout the semester during synchronous sessions.**

*****Presentations, interviews, and reports are submitted once after each three or four lectures throughout the semester during synchronous sessions.**

6. Course Syllabus

Chapter.	Subject	Content	Number of Learning Objects	Number of synchronous Learning Objects
CH1	Computer Program	1. The computer as a machine 2. H/W development & knowledge Democracy 3. Operating systems 4. Computers & operating systems 5. Classification of operating systems & their development 6. Data coding 7. Computers software 8. Programming languages 9. High level programming languages – historical background 10. High level programming languages – Procedural languages 1 11. High level programming languages – Procedural languages 2 12. High level programming languages – Functional languages 13. High level programming languages – Logical Languages 14. High level programming languages – Object oriented languages 15. Compilers 16. Exercises	31	15

		17. Systematic development of software 18. Software development strategies 19. Flowcharts 20. Algorithms 21. Pseudo code 22. Pseudo code basic instructions 23. Read instruction 24. Write instruction 25. Assign instruction 26. Conditional instruction 27. While instruction 28. Methodology of writing a software system 29. General classical examples 30. Exercises 31. Activity		
CH2	C# Basics	1. Microsoft Dot Net 2. Dot Net Framework 3. C# quick start 4. Analyze the script 5. C# Reserved words (Keyword) 6. Basic styles 7. C# variables 8. C# constants 9. Operations in C# and their preferences –1 10. Operations in C# and their preferences –2 11. Operations in C# and their preferences –3 12. Operations in C# and their preferences –4 13. Operations in C# and their preferences –5 14. Reading instruction	**15**	**7**

		15. Exercises for experimentation		
CH3	C# Instructions	1. General rules 2. Instructions blocks and variable range 3. Assign instruction 4. Conditional instruction 5. Conditional instruction ambiguity 6. Conditional assign instruction 7. While instruction 8. The five basic algorithm instructions in C# 9. Exercises 10. Problems 11. Issues to resolve algorithmically, and then by language C#	11	5
CH4	Algorithmic Language	1. Control instructions derived from the basic instructions 2. Loop Instruction for 3. C# frequency instruction 4. Loop Instruction for in C, C# 5. Examples of for Instruction 6. Loop instruction: Repeat one time at least 7. Example: Repeat one time at least 8. Example: Script do { } while 9. Break structured programming 10. Instructions to break structured programming in programming languages 11. Example of break instruction 12. Example of break instruction within for instructions block 13. The instruction continue in C, C#	17	8

		14. Branching instruction: multiple-choice conditional instruction 15. Branching instruction: switch … case 16. Programming examples: switch … case 17. Problems		
CH5	Structures of Composite Data	1. Composite data patterns 2. Character strings 3. Character strings – Declaring a string of characters 4. Character strings – the internal representation of the characters string and access to a character of the string characters 5. Character strings – Modification: Insert 6. Character strings – Modification: Merge using the "+" process 7. Character strings – Modification: Obtain the location of a partial string of the IndexOf character string 8. Character strings – Modification: Converts a string of characters to the ToCharArray character table 9. Character strings – Modification: Assign and comparision 10. Tables and matrices – definition of a table 11. Tables and matrices – Using tables and matrices 12. Tables and matrices – software examples 13. Tables and matrices – definition of a	13	6

		matrix, generating matrix cells whose dimensions are not defined at the definition		
CH6	Introduction to Functions and Procedures	1. The structure of the script in C# 2. Functions and procedures (methods) 3. Declare the method and define it in C# 4. Call a method 5. Passing Transactions – Introduction 6. Passing transactions – simple pattern homogeneity 7. Passing transactions – pass Value 8. Passing transactions – pass Address 9. Method result returning 10. Variables definition range 11. Row elements and procedures variables 12. Exercises for experimentation	12	6
CH7	Exercises and Problems	Exercises and problems 1. Exercise 1 2. Exercise 2 3. Exercise 3 4. Exercise 4 5. Exercise 5 6. Exercise 6 7. Exercise 7 8. Exercise 8	9	4

7. Practical Activity:

- Tools and Labs:

Tool Name	Description
Word, power point, excel	Microsoft office
Visual Studio	Microsoft https://visualstudio.microsoft.com/vs/preview/

- Practical Activities per Chapters:

Chapter	Practical Activity	Remarks
CH1	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects ☐ Experiments ☑ Discussion	
CH2	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects ☐ Experiments ☑ Discussion	
CH3	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects	

	☐ Experiments ☑ Discussion	
CH4	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects ☐ Experiments ☑ Discussion	
CH5	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects ☐ Experiments ☑ Discussion	
CH6	☑ Exercises ☑ Assignments ☑ Webinars ☐ Projects ☐ Experiments ☑ Discussion	
CH7	☑ Exercises ☑ Assignments ☑ Webinars ☑ Projects ☐ Experiments ☑ Discussion	

8. References:

- McGrath, Mike. (2016) "C# Programming in Easy Steps", Easy Step Limited, 138 pages

- w3schools.com (2019) "ASP.NET Web Pages - Examples in C# and VB" [online]. Available from: https://www.w3schools.com/asp/webpages_examples.asp

- John Sharp (2018) "Microsoft Visual C# Step by Step (Developer Reference)", 9th Edition, Microsoft Press, 832 pages

- Nakov, S. and Kolev, V. (2013) "Fundamentals of Computer Programming with C#: Programming Principles, Object-Oriented Programming, Data Structures", Faber, 1122 pages