



الجامعة الافتراضية السورية  
SYRIAN VIRTUAL UNIVERSITY

# Programming III

## Course Definition

**I**nformation

**T**echnology

**E**ngineering



Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

## 1. Basic Information:

Course Name	Programming III
Course Code	<b>BPG601</b>
Number of Presentational Sessions*	18
Number of Synchronous Sessions**	14
Number of Exams***	1
Theoretical Sessions Work Load (hrs.)	96
Practical Sessions Work Load (hrs.)	20
Credit Hours	5

\*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.).

\*\*\* The final exam is **1.25 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.

Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

## 2. Prerequisites courses:

Course	Code
Programming I, Programming II, Data Structures and Algorithms (1)	BPG401, BPG402, BDA501

## 3. Course Objectives:

Programming III is an integral part of Programming 1 and Programming 2:

It introduces advanced programming concepts in Python, which has been ranked for several years among the top five programming languages used by programmers. Consequently, software engineer must be able to use it, in a systematic and organized manner.

### New concepts include:

- Two-dimensional Arrays or Matrix (they are based on lists in Python).
- Subprograms (they are typeless in Python), and the lambda computation
- Collection data structures which are built in Python: **list, tuple, set, dictionary**
- Software packages / libraries

Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

#### 4. Learning Outcomes (LO):

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

1. Master basic programming in Python and knowing how it differs:
  - Basic instructions: **if**, **if-else**, **while**, **for**
  - Basic types: **int**, **float**, **bool**, **string**
2. Comprehend tables / lists / matrices and is proficient in programming them topics are:
  - The data structure of the table/array, and its corresponding in Python (list structure)
  - The importance of using tables in software applications
  - The importance of lists in Python as one of its strengths in writing applications in a concise and effective manner.
3. Understand strings and text files in Python and is proficient in programming them in his applications:
  - Strings: most important operations and methods on them.
  - Text files: write and read them
4. think back on the concept of subprograms/functions and everything related to them, and learns their specificity in Python, as functions are considered the most important strength in Python. Topics are:
  - Parameters Passing
  - Scope of variables
  - Defining anonymous methods using the lambda statement
  - Special functions: **filter** (), **map** (), **reduce** ()
5. know how to capture exceptions and handling them in Python

Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

6. Distinguish the types of collection data (**list, tuple, set, dict**) in Python, able to use it efficiently, being able to choose the appropriate type for the variables and data according to the problem that he wants to solve.
7. Know the Python OOP approach:
  - Building classes in Python
  - Defining data and functions members of a class
  - Using defined classes.
8. learn libraries modules and how to import them, and group libraries into software packages.
9. use the built-in functions and mathematical constants provided by the library **math** and can draw complex curves and scattered points using the **matplotlib** package and the **pyplot** library

Syrian Arab Republic	 SVU الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

## 5. Assessment Results:

Chapter Number	Chapter Title	General Objectives	Interactive Content & Recorded Sessions	Applied Activities Synch. Sessions	Final Exam ***
CH1	Basic Instruction and Data Types	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH2	Arrays-Lists	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH3	String and Text Files	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH4	Subprograms-Lambda function	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH5	Exceptions	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH6	Collection Data Types	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√
CH7	OOP	Comprehension	√	√	√
CH8	Packages	Comprehension	√	√	√
CH9	Packages: <i>math, matplotlib</i>	Comprehension Analytical Thinking Tools and Application Hands-On	√	√	√

\*\*\*The final exam is **1.25 hours** long and is given at the end of the course.

Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

## 6. Course Syllabus:

Chapter	Subject	Content	Number of Async Sessions	Number of Sync Sessions
CH1	Basic Instruction and Data Types	<ol style="list-style-type: none"> <li>1. –Basic instructions in Python</li> <li>2. –Primitive Types in Python</li> </ol>	2	2
CH2	Arrays–Lists	<ol style="list-style-type: none"> <li>3. Arrays one dimension (lists in Python)</li> <li>4. Arrays 2 Dimensions (Matrix)</li> </ol>	2	2
CH3	String and Text Files	<ol style="list-style-type: none"> <li>1. Strings: operations and functions</li> <li>2. Text files: read/write</li> </ol>	2	1
CH4	Subprograms– <i>lambda</i> function	<ol style="list-style-type: none"> <li>1. Function Declaration in Python</li> <li>2. Parameter Passing</li> <li>3. Internal Functions</li> <li>4. <i>lambda</i> Function</li> <li>5. Special functions: <b>filter(), map(), reduce()</b></li> </ol>	2	2
CH5	Exceptions	<ol style="list-style-type: none"> <li>1. Exceptions Catch</li> <li>2. Important Exceptions</li> </ol>	2	1
CH6	Collection Data Types	Operations and Functions defined On: <b>set, tuple, dic, list</b>	2	2
CH7	OOP	<ol style="list-style-type: none"> <li>1. Class declaration in Python</li> <li>2. Inheritance</li> <li>3. Access Specifiers</li> </ol>	1	1
CH8	Packages	<ol style="list-style-type: none"> <li>1. Libraries : modules</li> <li>2. packages</li> </ol>	2	1
CH9	Packages: <i>math</i> , <i>matplotlib</i>	<ol style="list-style-type: none"> <li>1. Mathematical Library: <i>math</i></li> <li>2. Graphical Drawing Package: <i>matplotlib</i></li> </ol>	2	2

Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

## 7. Practical Activity:

### 1. Tools and Labs:

Tool Name	Description
Spyder( python 3.x)	IDE for Python programming

### 2. Practical Activities per Chapters:

Chapter	Activities Type	Remarks
CH1	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)
CH2	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)
CH3	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)
CH4	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)
CH5	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)
CH6	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)



Syrian Arab Republic	 الجامعة الافتراضية السورية SYRIAN VIRTUAL UNIVERSITY	الجمهورية العربية السورية
Ministry of Higher Education		وزارة التعليم العالي
Syrian Virtual University		الجامعة الافتراضية السورية

CH7		Experimenting and explaining examples in IDE Spyder (python 3.x)
CH8		Experimenting and explaining examples in IDE Spyder (python 3.x)
CH9	<input checked="" type="checkbox"/> Exercises <input checked="" type="checkbox"/> Homework <input checked="" type="checkbox"/> Experiment	Experimenting and explaining examples in IDE Spyder (python 3.x)

## 8. References:

Python How to Program, Paul Deitel - Harvey Deitel,  
 Publisher Pearson, 2002

<https://www.tutorialspoint.com/python/>