

**1. General Information:**

Alpha-numeric codification: CIIC 3011  
Course Title: Introduction to Computer Programming I  
Number of credits: 3  
Contact Period: 3 hours of lecture per week

**2. Course Description:**

**English:** Analysis of algorithmic problems, development of solutions, and their implementation in a high level programming language using object-oriented programming techniques. Topics: Numerical systems, internal representation, constants, variables, and data types, selection, and iteration control structures, functions, and data passing mechanisms, basic data structures, pointers, and dynamic memory management, data input/output, files, and software development environments.

**Spanish:** Análisis algorítmico de problemas, desarrollo de soluciones, y su implementación en un lenguaje de programación orientado a objetos. Temas: sistemas numéricos, representación interna, constantes, variables, tipos de datos, estructuras de control e iteración, funciones, métodos para paso de parámetros, estructuras de datos básicas, apuntadores, archivos, y sistemas de desarrollo de software.

**3. Pre/Co-requisites and other requirements:**

None

**4. Course Objectives:**

Students will get an overview of the computing discipline by studying introductory, but fundamental, topics that are needed for future courses in the program and by every professional in the discipline. Students will become proficient in the development of computer programs using object-oriented techniques.

**5. Instructional Strategies:**

conference    discussion    computation    laboratory  
 seminar with formal presentation    seminar without formal presentation    workshop  
 art workshop    practice    trip    thesis    special problems    tutoring  
 research    other, please specify: \_\_\_\_\_

**6. Minimum or Required Resources Available:**

Students will use the Departmental computer laboratories to complete course projects.

**7. Course time frame and thematic outline**

Outline	Contact Hours
Introduction to Computing and algorithms	5
Software Development Process and environments, mobiles apps and pair programming	4
Abstraction and data types and Representation	4
Operations and Expressions	4
Analyzing problems and algorithms	4
Functions	4
Objects and Classes	4
Files and I/O Streams	3
Arrays and Vectors	2
Classes and ADT	4
Pointers and Dynamic Memory	4
Exams	3
<b>Total hours: (equivalent to contact period)</b>	<b>45</b>

**8. Grading System**

<input checked="" type="checkbox"/> Quantifiable (letters) <input type="checkbox"/> Not Quantifiable
--

**9. Evaluation Strategies**

Activity	Quantity	Percent
<input checked="" type="checkbox"/> Exams	3	40%
<input checked="" type="checkbox"/> Final Exam	1	25%
<input type="checkbox"/> Short Quizzes		
<input type="checkbox"/> Oral Reports		
<input type="checkbox"/> Monographies		
<input type="checkbox"/> Portfolio		
<input checked="" type="checkbox"/> Projects	variable	35%
<input type="checkbox"/> Journals		
<input type="checkbox"/> Other, specify:		
<b>TOTAL:</b>		<b>100%</b>

**10. Bibliography:**

1. Tony Gaddis, *Starting out with C++*, 7th ed., Addison Wesley, 2011.
2. Tony Gaddis, *Starting out with Python*, 2<sup>nd</sup> ed., Addison Wesley, 2012.
3. John Zelle, *Python Programming: An Introduction to Computer Science*, 2nd ed., Franklin, Beedle & Associates, 2010.
4. *Mobile CSP - An AP Computer Science Principles*
5. *Blown to Bits - Your Life, Liberty, and Happiness After the Digital Explosion*
6. Hal Abelson Ken Ledeen Harry Lewis
7. Diane Zak, *An Introduction to Programming With C++*, 7th ed., Cengage Learning, 2012.

**11. Course Outcomes**

Upon completion of this course the student will be able to:	Program Outcomes
1. explain the meaning of the discipline of computer science, basic computer terminology and the impacts of computing in the society	4
2. explain the wide variety of computational artifacts that can be created using software tools and their impact in modern society	4
3. apply programming language concepts; such as, variables, data types, flow control structures, memory management principles, memory representation of data, files, and the compiling process	6
4. use software to solve problems: including basic algorithmic, data structures and problem-solving techniques (decision structures, loops, functions), strategies for structuring and reusing code to solve problems	6
5. apply contemporary software development and management tools (design, documentation, implementation and testing solutions) to design and develop software applications.	6

**12. According to Law 51**

Students will identify themselves with the Institution and the instructor of the course for purposes of assessment (exams) accommodations. For more information please call the Student with Disabilities Office which is part of the Dean of Students office (Office #4) at (787)265-3862 or (787)832-4040 extensions 3250 or 3258.

**13. Academic Integrity**

-The University of Puerto Rico promotes the highest standards of academic and scientific integrity. Article 6.2 of the UPR Students General Bylaws (Board of Trustees Certification 13, 2009-2010) states that academic dishonesty includes, but is not limited to: fraudulent actions; obtaining grades or academic degrees by false or fraudulent simulations; copying the whole or part of the academic work of another

person; plagiarizing totally or partially the work of another person; copying all or part of another person answers to the questions of an oral or written exam by taking or getting someone else to take the exam on his/her behalf; as well as enabling and facilitating another person to perform the aforementioned behavior. Any of these behaviors will be subject to disciplinary action in accordance with the disciplinary procedure laid down in the UPR Students General Bylaws.—