

Universidad de Puerto Rico Mayagüez

CIIC 4010/ICOM 4015 : Programación Avanzada Primer Semestre 2016-2017¹ Prontuario del Curso

Profesor:
Bienvenido Vélez, PhD
Horas de Oficina: MJ 3:15-4:45PM
Oficina: S-219 y S-701
Email: Bienvenido.Velez@upr.edu

Asistentes de Cátedra:
Paul Bartus y Leo Solorzano

Introducción

En este curso se estudiarán técnicas avanzadas de programación de computadoras, con énfasis en la programación orientada a objetos y el desarrollo de estructuras de datos. Se asume que el estudiante tiene conocimientos previos en el área de computación, al menos al nivel de lo que se cubre en los cursos CIIC 3011 ó INGE 3016². Estaremos usando el lenguaje *Java*. En general estudiaremos sobre el uso de clases, programación orientada a objetos, estructuras de control, referencias, arreglos, clases, conceptos de estructuras de datos, conceptos importantes de la recursión, su implementación y su aplicación al desarrollo de algoritmos y aplicaciones. Estudiaremos sobre herencia, interfaces y "method overloading". Se estudiarán distintos problemas clásicos (por ejemplo, búsqueda y ordenamiento) y soluciones comunes, además de conceptos básicos de estructuras de.

Objetivos

Los temas y tareas con los que nos involucremos en este curso le permitirán, entre otras cosas:

1. Conocer en términos generales lo que representa la disciplina de computación con sus áreas fundamentales y familiarizarse con la jerga de ésta.
2. Exponernos al tipo de problema que se estudian en cada una de estas áreas y experimentar con el desarrollo de programas aplicados a los mismos.
3. Conocer las operaciones fundamentales que puede llevar a cabo la computadora, lo que implicará, implícita o explícitamente, el tipo de instrucción que se puede esperar en un algoritmo a ser ejecutado por la computadora. Esto lo veremos, en la mayoría de los casos mediante el estudio del efecto de cada instrucción del lenguaje Java.
4. Poder entender programas escritos en el lenguaje Java y, con mínimo esfuerzo, con la ayuda de manuales o textos, poder entender otras instrucciones y partes del lenguaje no discutidas en clase: sus reglas gramaticales y su efecto al ser ejecutadas por la computadora (lo que ésta hace cuando las ejecuta).
5. Poder desarrollar algoritmos para resolver problemas de complejidad mediana en los que se requieran estructuras de datos simples y técnicas básicas de programación, y codificar dichos algoritmos usando el lenguaje Java.
6. Conocer sobre aplicaciones comunes en las que la computadora es de gran utilidad y distintas técnicas para trabajar con problemas relacionados: (por ejemplo, ordenamiento y búsqueda de datos).
7. Conocer sobre la importancia que tiene la profesión de Ingeniero/a de Computadora en la sociedad moderna y sobre las reglas de comportamiento ético que deben seguirse al ejercer la misma.

¹ Preparado por Pedro I. Rivera Vega. Actualizado por Bienvenido Vélez.

² Es requisito el haber aprobado el curso INGE 3016 ó CIIC 3011 para poder estar matriculado en CIIC4010/ICOM 4015.

Lenguaje de Programación a Estudiar

El lenguaje de programación que estaremos estudiando es el lenguaje Java. El aprender un lenguaje de programación en la clase se hace con el propósito de tener una herramienta que nos permita poner en práctica todos los conceptos a estudiar que son esenciales para el desarrollo de algoritmos e implementación de los mismos en la computadora. Y al aprender ese lenguaje estaremos a la vez aprendiendo muchos de esos conceptos. El lenguaje a estudiar es un lenguaje de uso común en muchas aplicaciones reales hoy día. Es también uno de los lenguajes de mayor uso en cursos como éste y en cursos posteriores en otras universidades. Todo lo que estudiemos en la clase en cuanto al desarrollo de algoritmos y programas lo estaremos viendo en base a este lenguaje de programación. Los conceptos a estudiar en clase son válidos también en la mayoría de los lenguajes de programación que se usan hoy día y se aplican de manera similar en la mayoría de ellos.

Evaluación

La evaluación en este curso consiste de exámenes, trabajos de laboratorio y asignaciones para entregar (proyectos de programación). La nota final que obtenga el estudiante estará basada en su desempeño en las distintas actividades de evaluación. En cada actividad el/la estudiante obtiene una puntuación final de 0 a 100. Cada una de esas puntuaciones finales por actividad se usa para calcular el promedio final aplicándole el peso que se ilustra en la siguiente tabla.

Actividad	Peso en el Promedio Final ³
Exámenes Parciales (3)	45%
Proyectos de Programación	15%
Laboratorio	15%
Asistencia a Clases y Participación ⁴	05%
Examen Final	20%

Curva Final

Su nota final se calculará basada en el promedio final, aplicándole la siguiente curva:

Promedio Final ⁵	Nota
[90-100]	A
[80-90)	B
[65-80)	C
[55-65)	D
[0-55)	F

IMPORTANTE: Para poder aprobar este curso, se requiere nota final de C ó más según los pesos establecidos en la tabla, pero también es necesario que obtenga puntuación final de al menos 50 en las actividades correspondientes a laboratorio y al menos 50 en las actividades correspondientes a proyectos.

³ Los pesos aquí enunciados podrían variar levemente dependiendo de alguna situación inesperada que tenga que ser considerada. De hacerse, se aplicaría a todos los estudiantes por igual y tiene que cumplirse que, para todo estudiante, la puntuación final bajo los nuevos pesos no puede ser menor que la puntuación final que el estudiante pueda obtener usando los pesos aquí anunciados.

⁴ La nota parcial final en este renglón se calcula usando la siguiente regla. La puntuación a obtener por este concepto será un valor entre 0 y 100. Por cada ausencia registrada luego del primer día de clases perderá 20 puntos.

⁵ La puntuación final se redondea al entero más cercano.

Exámenes

En este curso se ofrecerán **tres exámenes parciales** en los días que indicamos a continuación, en el horario de su laboratorio correspondiente. Las fechas para los exámenes se detallan en la siguiente tabla:

Examen	Fecha	Salón	Hora
1	14 de septiembre de 2016	S-121	hora de laboratorio
2	19 de octubre de 2016	S-121	hora de laboratorio
3	16 de noviembre de 2016	S-121	hora de laboratorio

Debido a la naturaleza del curso, **todos los exámenes son acumulativos**, requiriendo material cubierto desde el comienzo del curso hasta el momento del examen. Pero el énfasis en cada caso se dará al material cubierto desde el examen previo. Cada examen parcial tendrá una duración de **dos horas** y enfatiza en el material presentado en la tabla que desglosa la secuencia de tópicos a discutir. Este material puede variar, pero en tal caso se les notificará con suficiente tiempo. El **examen final** incluirá preguntas relativas a **todo el material del curso** y se ofrecerá en fecha y horario a decidir por las autoridades pertinentes.

De usted tener ya algún curso que se ofrece en alguna de las horas anunciadas para examen en este curso, o en el que ya se le ha anunciado (tiene que haber sido primero que nosotros) algún examen o actividad requerida que conflija con alguno de estos exámenes, tiene que notificarlo tan pronto termine la primera clase. Luego de esto no se aceptarán excusas de ese tipo, puesto que se entiende que cualquier actividad que le haya sido anunciada se habría hecho luego de yo haber anunciado nuestros exámenes, por lo que entonces se debe entender que tenemos prioridad.

Proyectos de Programación

En este curso se asignarán varios proyectos de programación. En cada caso se le especificará una tarea, la cual, una vez completada, implementará un sistema con la funcionalidad que finalmente se persiga. Para cada proyecto van a recibir un documento que se les entregará con al menos dos semanas de anticipación a la fecha en que se vence el mismo. Ese documento ha de contener las especificaciones del proyecto particular: lo que tiene que hacer, la manera en que debe entregar, lo que tiene que entregar y la fecha de entrega. Es importante recalcar que estos trabajos tienen que entregarse a tiempo, en o antes de la fecha establecida en cada caso. En cada caso se les indicará la penalidad que se estará adjudicando como consecuencia de entregar un trabajo tarde.

También es importante que tenga en cuenta que **el trabajo en cada proyecto debe ser individual**, excepto si en las especificaciones del proyecto mismo se establece otra cosa en alguna que otra parte del mismo. **Con respecto a esto es importante que tenga en cuenta que de tener evidencia de que alguien se ha copiado alguna parte del proyecto (o un examen) de alguna fuente (otro estudiante, Internet, etc.), todos los que hayan participado en dicha actividad se corren el riesgo de ser sometidos a sanciones ya establecidas en los reglamentos de la Universidad de Puerto Rico. No se corra el riesgo!!!**

Recuerde, usted está tomando este curso porque quiere aprender y porque lo que aprenda aquí lo necesitará en cursos posteriores de su programa académico, particularmente en aquellos en que se requiere del desarrollo de software, y en su trabajo futuro como Ingeniero/a de Computadoras. La experiencia muestra que **leer y trabajar** en los detalles de los proyectos, laboratorios y ejercicios asignados es la mejor manera de aprender. Usted notará que, si lo hace así, ya estará estudiando para los exámenes y que tendrá entonces grandes posibilidades de salir bien en ellos. **En los exámenes habrá preguntas relativas a las etapas del proyecto, ejercicios de laboratorio y otros ejercicios asignados.**

Usted no debe limitar su uso del centro de cómputos a trabajar en las asignaciones para entregar, puede utilizarlo en cualquier momento que le sea conveniente para ayudar a entender instrucciones y conceptos estudiados en clase. De hecho, esto último es altamente recomendable pues es una de las mejores formas de aprender.

Importante: En los casos en que se entienda necesario, **la corrección de un proyecto puede incluir una entrevista individual con cada estudiante.** En dicha entrevista se les pueden hacer preguntas relativas a su solución y a cómo extendería la misma para lograr alguna otra funcionalidad no necesariamente especificada, pero que el/la corrector/a entienda que debe poderse hacer sin mucha dificultad. La puntuación que usted obtenga en cada etapa del proyecto entregado les será notificada en aproximadamente dos semanas a partir de la fecha límite establecida para la entrega de la misma. Se incluirá descripción de las razones por las cuales ha perdido puntos (si ese es el caso). La entrega de estos resultados se hará durante la hora de la clase. Luego de esa fecha en la que se entregan los resultados en la clase usted tendrá la oportunidad de hacer reclamaciones durante las **dos semanas siguientes**⁶. Si usted no está presente al momento en que se entreguen los resultados de un proyecto, es responsabilidad suya el reclamar sus resultados durante horas de oficina. **Recuerde, el periodo de dos semanas para hacer reclamaciones comienza a partir del día en que se entregan los resultados en la clase, independientemente de si usted no haya recibido los mismos por razones de ausencia.** Sus notas también serán registradas en el eCourses (Moodle).

Laboratorio

El curso incluye una sesión de laboratorio, la cual se ofrece una vez a la semana (vea su programa de clases). **La asistencia al mismo es obligatoria.** En este laboratorio se estarán poniendo en práctica distintos conceptos estudiados en la clase, además de introducir conceptos importantes sobre el uso de las herramientas de desarrollo de programas. En una sesión de laboratorio usualmente se asignan ejercicios relacionados al tema en discusión, los cuales tiene que entregar funcionando correctamente. Los mismos serán corregidos y acumulados para su nota final en esta parte del curso. Recibirán más detalles de esto en el laboratorio mismo.

Asistencia a Clases y Participación

Durante algunas clases se proyectarán en pantalla ejemplos de programación que el profesor realizará en su computadora. Es aconsejable que los estudiantes tengan sus computadoras personales disponibles en clase para realizar los ejercicios discutidos en sus computadoras individuales de forma que se facilite al aprendizaje y se puedan atender las preguntas o problemas que surjan en el momento.

La asistencia a clases, así como al laboratorio (según lo establece su programa de clases) es obligatoria. Note que la misma tiene un peso de un **5%** en el cómputo del promedio final, pero tenga en cuenta la regla que menciono adelante con respecto a más de seis ausencias. La nota parcial final en este renglón se calcula usando la siguiente expresión: $\max(100 - x * 20, 0)$, donde **x** es el número de ausencias. El primer día de clases (16 de agosto 2016) no se cuenta. Note que 5 ó más ausencias representará una puntuación de 0 en esta categoría. En cada clase se verificará asistencia al inicio de la misma usando la lista de estudiantes de la sección particular solamente. Cualquier persona que llegue a la clase luego de haber completado esa verificación inicial deberá notificarlo, en cuyo caso se le concederá una **tardanza**, en vez de una ausencia. Sin embargo, cada **dos tardanzas** se cuentan como **una ausencia**. Finalmente, **de usted tener 7 (siete) ausencias registradas o más durante el semestre, sin una razón de peso que pueda ser corroborada, su nota en el curso será F automáticamente.**

De usted tener que salir del salón durante el tiempo que dure la conferencia, debe entonces notificar su razón (sin tener que entrar en detalles) al final de la clase. De no hacerlo, la salida se le contará como una ausencia. Al final de la clase se podrá leer la lista de aquellos que han firmado para corroborar su presencia.

⁶ Excepto en el último proyecto, pues el mismo se corrige durante el tiempo de exámenes finales, y muy cercano a la fecha límite para la entrega de notas. Además este periodo de tiempo es final y firme e independiente de si usted está presente en la clase al momento de entregar el trabajo corregido. Esta regla también aplica a exámenes corregidos y entregados en clase.

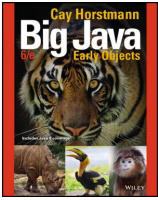
Si usted tiene que ausentarse de alguna clase por razones de peso, tendrá que mostrar evidencia en las horas de oficina para que esa ausencia no se le cuente. **Bajo ninguna circunstancia se aceptará como excusa válida el que usted se ausente a alguna sesión de nuestras clases o del laboratorio por razón de tener que tomar un examen (o alguna actividad de ese tipo) de alguna otra clase en alguna hora que coincida con las asignadas a este curso.**

OTRAS REGLAS QUE TIENE QUE MANTENER:

- **No se permite el uso de teléfonos celulares** durante la clase. Mantenga el mismo apagado o en modo silencioso durante la clase. Si por casualidad el mío suena, recuérdeme esta regla...
- **No se permite el conversar con compañeros/as** en voz alta en el salón durante la clase, excepto cuando se establece una actividad de ese tipo durante la clase o en aquellos casos en que se requiera para satisfacer algún requerimiento de acomodo razonable.
- **Levante su mano** siempre que quiera hacer una pregunta o algún comentario sobre lo que se esté discutiendo en la clase.
- En todo momento tiene que cumplir con las reglas o regulaciones que establece la UPR a través de certificaciones oficiales y reglamentos. Además, es necesario cumplir con todas las leyes que aplican a residentes de Puerto Rico.
- Lea, piense, haga ejercicios (tanto del material de la clase, así como ejercicios físicos...)

Libro de Texto

El libro de texto para este curso es el siguiente:

	<p>Big Java : Early Objects (6th Edition) Cay Horstmann John Wiley ISBN 978-1-119-14159-4</p>
------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------

Referencias

En general, cualquier libro de referencia al lenguaje Java o de introducción a la computación le debe ser de utilidad como libro de referencia para este curso. También debe conseguirse un libro sobre el sistema Unix o Linux, ya que ese es el sistema que estaremos usando en el laboratorio. De hecho, libros en esta categoría ya no deben faltarle en su biblioteca privada...

Daremos información sobre libros específicos más adelante en el curso, pero échele un vistazo a las páginas siguientes en internet:

- Página del autor del texto: www.horstmann.com
Es necesario que se conecte a esta página. En la misma encontrará información muy valiosa sobre el contenido del texto, código de ejemplos que se discuten en el texto, presentación de los tópicos del curso en formato html, cómo usar el sistema Java, etc.
- Página del texto: www.horstmann.com/bigjava.html
- Página oficial de Java: java.sun.com
- Un tutorial de Java: docs.oracle.com/javase/tutorial/java/

ICOM 4015. Topic Outline and Schedule

The following pages detail the topics to be covered in this course, day by day, throughout the current semester. Be aware that this is an approximate agenda; it might happen that we can move faster on some topics, but slower than estimated on others. The estimated time on some topics is based on the fact that you should already have some knowledge of the programming concepts and constructs using a similar-to-Java programming language (at least, what you should have studied in CIIC 3011 or INGE 3016). It may also be the case that some planned topic cannot be covered because of time. However, we will follow this agenda the best we can.

The agenda is based on the material in the textbook. You can always access useful tools and information at the author's website: <http://www.horstmann.com/>. You should get used to examine the additional material at <http://bcs.wiley.com/he-bcs/Books?action=index&itemId=1118431111&bcsId=7872>, which is the Student Companion Site for the textbook. Several web sites provide useful additional material about the discipline of Computer Science and similar disciplines; for example, take a look at the following: <http://www.computersciencestudent.com/>. This last one has many resources that are helpful not only for this course, but as an engineering student in general; so, get used to use it.

In terms of resources where to look for descriptions about the Java language, perhaps one of the best is the following: <http://docs.oracle.com/javase/tutorial/index.html>. In particular, you should first look for "Trails Covering the Basics" and "Creating Graphical User Interfaces".

You need to get used to read from the textbook and take advantage of all the alternative learning strategies and tools that the author has incorporated into it. Carefully study each example being discussed, understand what is being asked, and understand the solution being discussed. Throughout each chapter in the textbook you will find different simple "self check" questions. Those are simple questions that you should be able to determine a simple answer based on the material being covered on the preceding pages. At the end of each chapter, you will find several exercises grouped in three categories:

- Review Questions - These are usually exercises with simple answers, usually directly related to the explanations given in the particular chapter. They not necessarily ask for programming solutions.
- Practice Exercises- These are problems in which it is expected that you provide a Java solution based on whatever constructs about this programming language have been discussed in the chapter or earlier chapters in the textbook.
- Programming Projects - These are usually harder problems that need to go through the whole design process so as to be able to produce a programed or algorithmic solution.

You should at least read over those exercises and solve as much as you can on each category. Try at least to understand what is being asked and think about solutions. You are expected to at least answer each of the "review questions" and most of the "practice exercises". The more time you put on this, the better you will understand the material being discussed and less will be the chances of easily forgetting them.

Tentative Outline of the Course – List of topics by Lecture⁷

Lecture Number	Topic or Activity
1	General discussion about the course: the main objectives, the textbook, evaluation criteria, laboratory sessions, contents of the course and of each exam, dates of exams (out of regular hours of the course), and general idea of programming projects.
1-2	Chapter 1: Introduction to computing (most of this material should be a general review for all of you - remember INGE 3016), introduction to the Java system , and how to get (download) it free... visit http://java.sun.com . General format of a Java program, compilation and execution of Java program, and recognition of errors. Suggested Exercises: R1.1 – R1.18, E1.1 – E 1.7, E1.14 - E1.17
3-4	Chapter 2: Using Objects. Introduction to the use of objects and classes in Java. Declaration of variables to reference objects, operating with objects. Methods, parameters, return values, and application of methods to object. API documentation. Introduction to graphics in Java. Suggested Exercises: R2.1 – R2.25, E2.4 - E2.9, E2.11 - E2.13, P2.1, P2.4, P2.7, P2.11, P2.12
5-7	Chapter 3: Implementing Classes. How to implement classes in Java, implementation of simple methods, constructors, instance variables, and local variables. Classes for graphical shapes. Suggested Exercises: R3.1 – R3.22, E3.1 - E3.3, E3.8, E3.11, E3.16 - E3.18, P3.9
8	Chapter 4: Fundamental Data Types. The numeric data types, strings, characters, and booleans in Java. Variables, constants and literals of each of these primitive data types. The assignment statement, mathematical functions, type conversions, reading input, and comparison between primitive data types and objects. The String data type. Formatted I/O in Java. Suggested Exercises: R4.1 – R4.23, E4.5, E4.7, E4.10, E4.12, E4.13, E4.15, E4.18, E4.20, E4.23, P4.2, P4.3, P4.5 - P4.8, P4.10, P4.12
EXAM	Exam 1 : February 22, 2016 : during your lab session
9	Chapter 5: Decisions. Review of the topic of decisions in programs, this time in the context of Java: selection constructs in Java, the <i>if-statement</i> and the <i>switch-statement</i> . We also study about Boolean expressions, grouping of statements in a program, and important issues about testing computer programs. Suggested Exercises: R5.1 – R5.32, E5.1, E5.4, E5.9, E5.15, E5.17, E5.24, P5.2, P5.10
10-11	Chapter 6: Iteration. Review of the topic of iterations in programs in the context of Java. In particular, we study the <i>while</i> and for <i>loops</i> in Java. We apply them in processing input and in computer simulations. Debugging techniques and tools are also studied. Suggested Exercises: R6.1 – R6.29, E6.1 – E6.4, E6.19, P6.1 – P6.4, P6.7, P6.17, P6.18
11-12	Chapter 7: Arrays and Array Lists. Introduction to arrays and array lists in Java. Wrapper classes, auto-boxing, the enhanced loop, and two-dimensional arrays. Different applications and typical operations with these structures are discussed. Suggested Exercises: R7.1 – R7.33, E7.1 – E7.5, E7.10 – E7.12, E7.14, E7.20, P7.3, P7.7, P7.12
13-15	Chapter 8: Designing Classes. We discuss about the process of discovering classes during the analysis of program specifications, implementation by properly assigning attributes, pre-conditions and post-conditions, and definition of methods correctly. We will also study about static methods, variables, and how to use packages. Suggested Exercises: R8.1 – R8.26, E8.1 – E8.6

⁷ Basado en periodos de 75 minutos. El tiempo separado para cada t3pico es algo aproximado, el mismo puede variar durante el transcurso del semestre dependiendo de las necesidades que surjan. Note que la secuencia de periodos solo incluye conferencias, no los ex3menes. Adem3s, el n3mero de 3stos deja algunos espacios disponibles para ex3menes, repaso, discusi3n de ex3menes, discusi3n de proyectos, etc. Cada secci3n correspondiente del libro de texto queda asignada para su lectura y llevar a cabo actividades de pr3ctica; al menos los ejercicios sugeridos en cada cap3tulo..

EXAM	Exam 2 : March 21, 2016 : during your lab session
16-17	Chapter 9: Inheritance. The concept of inheritance in object oriented programming and how it is achieved in Java. Suggested Exercises: R9.1 - R9.11, E9.1 – E9.14, P9.1 - P9.3
18-19	Chapter 10: Interfaces and Polymorphisms. Interfaces in Java, their applications in defining reusable code, classes the implement interfaces. We also study the concept of super-type and subtype, and conversions between them, polymorphisms, the use and implementation of inner classes, and the implementation of listeners for timer events. Suggested Exercises: R10.1 - R10.22, E10.1 - E10.23, P10.1 - P10.11
20-21	Chapter 11: I/O and Exception Handling. We will learn how to write Java programs that interact with disk files (I/O) and other sources of bytes and characters. We will also study data encryption as an application in which streams are useful (If time permits...) and serialization of objects. We also study the concept of program exceptions, how to create them in a Java program, and how, when and where to catch them in the program. Suggested Exercises: R11.1 - R11.17, E11.1 - E11.14, P11.1 - P11.9
22-23	Chapter 12: Object-Oriented Design. Discussion of useful techniques and strategies to the design and implementation of software systems to handle non-trivial applications following the object-oriented approach. We discuss principles of UML, and relationship among classes. Suggested Exercises: R12.1 - R12.15, E11.1 - E11.6, P12.6 - P12.8
EXAM	Exam 3 : April 18, 2016 : during your lab session
24-25	Chapter 13: Recursion. Deep study of the concept of recursion and the way it is implemented with the activation record and process stack. We will see applications in which recursion represents a natural solution, compare it with iteration, and how to use helper or auxiliary recursive methods to solve problems. Suggested Exercises: R13.1 - R13.11, E13.3, E13.7 - E13.9, E13.11
26-27	Chapter 14: Sorting and Searching. Sorting and searching are probably the most common tasks being performed in computer applications. Here we will study different algorithms to do both operations on arrays or lists. We will have a chance to study different algorithms to accomplish both tasks and compare them, hence having an introduction into the important field of “analysis of algorithms”. Suggested Exercises: R14.1, R14.2, R14.10, R14.12, R14.13, R14.15, R14.17, E14.1 - E14.3, E14.5 - E14.10, E14.13, E14.14, E14.15.
EXAM	Final Exam : Date and Place To Be Announced by Registrar

Si busca usando "search engine" (Ejemplo: Google, Bing, Reddit, ...) por tópicos como "java tutorial", o algo de ese tipo, va a encontrar muchos otros lugares en el internet en donde hay información de utilidad sobre el lenguaje Java.

Reglamento de la Universidad sobre Integridad Académica

El artículo 10 del Reglamento General de Estudiantes de la Universidad de Puerto Rico contiene 15 puntos que se consideran "infracciones de las normas esenciales al orden y a la convivencia universitaria y acarrear sanciones disciplinarias." He aquí uno de los puntos.

La obtención de notas o grados académicos valiéndose de falsas y fraudulentas simulaciones, o haciéndose pasar por otra persona, o mediante treta o engaño, o copiando total o parcialmente la labor académica de otro estudiante, o copiando total o parcialmente las respuestas de otro estudiante a las preguntas de un examen, o haciendo o consiguiendo que otro tome en su nombre cualquier prueba o examen oral o escrito.

Violaciones a estos puntos pueden conllevar algunas de las siguientes sanciones:

1. *Amonestación*
2. *Probatoria por un tiempo definido durante el cual otra violación de cualquier norma tendrá consecuencia de suspensión o separación*
3. *Suspensión de la Universidad por un tiempo definido. La violación de los términos de la suspensión conllevará un aumento del período de suspensión o la separación definitiva de la Universidad.*
4. *Separación definitiva de la Universidad.*

Obtener información del Internet no autorizada por el profesor será considerado una violación flagrante del estatuto anterior. El estudiante que viole este reglamento obtendrá F en la clase y su caso podrá ser llevado ante la junta de disciplina del Recinto. Evítese este mal rato, o atégase a las consecuencias.

Esperamos no tener que lidiar con una situación como esta, pero estamos preparados para responder enérgicamente de ser necesario.

De encontrar algún error o inconsistencia en el contenido de este documento, favor de notificarlo.