Universidad de Puerto Rico
Recinto Universitario de Mayagüez
Departamento de Ingeniería Eléctrica y Computadoras

Curso ICOM 4015: Programación Avanzada
Primer Semestre 2013-2014

Prof. Bienvenido Vélez[1]
Oficina: Stefani 701
Teléfono: 832-4040 x5827
Horas de Oficina: Martes y Jueves 3:00-5:00PM o por acuerdo
E-mail: Bienvenido.Velez@upr.edu
Web: http://www.ece.uprm.edu/~bvelez
Conferencia: Martes y Jueves 5:00 pm - 6:15 pm Salón S 113

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 el curso INGE 3016[2]. Estaremos usando el lenguaje *Java*. En general estudiaremos sobre el uso de subprogramas, 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. Veremos 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 datos y estructuras de datos básicas.

Objetivos

Los temas y tareas con los que nos envolveremos 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 estudia en cada una de estas áreas y experimentar con algunos mediante el desarrollo de programas para resolver 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 de programación básicas, 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).

[1] Prontuario modificado de uno originalmente creado por el Prof. Pedro I. Rivera Vega

[2] Es requisito el haber aprobado el curso INGE 3016 para poder estar matriculado en ICOM 4015

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.

Lenguaje de Programación a Estudiar

El lenguaje que estaremos utilizando para desarrollar nuestros programas 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 mecanización de los mismos utilizando la computadora. Al aprender ese lenguaje estaremos a la vez poniendo en práctica muchos de esos fundamentos. 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. Aprender a programar en otros lenguajes similares simplemente requerirá, en la mayoría de los casos, de un manual o texto que nos explique el formato de las instrucciones equivalentes en dicho lenguaje.

Evaluación

La evaluación en este curso consiste de exámenes, trabajos de laboratorio y asignaciones de programación. La nota final que obtenga el estudiante estará basada en su desempeño en las distintas tareas de evaluación. En cada tarea el/la estudiante obtiene una nota parcial final. Cada una de estas tareas se refleja en la nota final de acuerdo al peso que se ilustra en la siguiente tabla.

Actividad	Peso en el Promedio Final
Exámenes Parciales (3)	45%
Asignaciones de Programación	20%
Laboratorios	10%
Examen Final	25%

Las puntuación en el renglón de asignaciones de programación podrá ser suplementada por un bono de hasta un 20% por concepto de promedio de calidad de sus ejecutorias según evaluadas de forma confidencial por sus compañeros de equipo en aquellos proyectos donde el trabajo no sea individual. La sección de Asignaciones de Programación contiene más detalles sobre este tema.

Exámenes

Debido a la naturaleza del curso, **todos los exámenes son acumulativos**, requiriendo material cubierto hasta el momento del examen. Pero el énfasis en cada caso se dará al material cubierto desde el examen previo. **Se ofrecerán tres exámenes parciales en días y horarios que serán avisados con al menos dos semanas de anticipación.** Cada examen parcial tendrá una duración de no menos de 75 minutos y enfatizará en el material presentado en clase y discutido en los textos y demás materiales distribuidos durante el curso. 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.

Los exámenes parciales serán tomados en línea durante la hora de laboratorio utilizando el sistema Moodle y usualmente incluirán una parte práctica donde el estudiante trabajará un problema de programación utilizando las herramientas de desarrollo provistas para el examen. Las soluciones que envuelvan programación no serán aceptadas ni corregidas si las mismas no compilan correctamente y/o no pasan las pruebas mínimas requeridas en cada caso.

Importante: Tomar todos los exámenes parciales y el examen final es requisito indispensable para aprobar el curso. Si usted tiene que ausentarse por causa fortuita de peso debe informar al profesor ANTES de la fecha del examen.

Asignaciones de Programación

Las asignaciones de programación son un componente pedagógico fundamental del curso ya que las mismas le permitirán al estudiante poner en práctica y demostrar sus capacidades en las destrezas de ingeniería de software requeridas en cursos posteriores en ingeniería de computadoras.

En cada asignación se le especificará una tarea, la cual, una vez completada, implementará un sistema con la funcionalidad que finalmente se persiga. Para cada proyecto los estudiantes recibirán un documento que se les entregará con al menos dos semanas de anticipación a la de entrega. Ese documento ha de contener las especificaciones del proyecto particular: lo que tiene que hacer, la manera en que debe entregarlo, 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 tardíamente.

Las asignaciones de programación podrán ser realizadas individualmente o en equipos de dos estudiantes. En caso de una asignación en equipo el estudiante podrá seleccionar a su compañero de equipo, per deberá seleccionar a un compañero distinto para cada asignación. Los miembros de un equipo deberán trabajar en estrecha colaboración de forma que ambos entiendas a la perfección todo el trabajo realizado para completar la asignación. Cada estudiante tendrá la oportunidad de evaluar la ejecutoria de su compañero de cada proyecto. El promedio de las evaluaciones de sus compañeros normalizadas a una escala de 1 al 10 constituirá una bonificación de hasta un 20% para el renglón de asignaciones de programación.

Importante: Entregar todas las asignaciones de programación funcionando correctamente es requisito indispensable para poder aprobar el curso. Asignaciones con errores de compilación no serán aceptadas.

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. Fomentamos la colaboración y solidaridad entre estudiantes siempre y cuando la misma no conlleve compartir código fuente o "source code".

Tenga en cuenta que de tener evidencia de que alguien ha copiado alguna parte de una asignación de programación de alguna fuente (otro estudiante, Internet, etc.), todos los que hayan participado en dicha actividad fracasarán automáticamente el curso y podrán ser sometidos a las 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 es esencial para poder tener éxito en cursos posteriores del programa ICOM, particularmente en aquellos en que se requiere del desarrollo de software, y en su carrera profesional. 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, el estudiar para los exámenes se le hará mucho más fácil. **Los exámenes podrán incluir preguntas relativas a cualquier faceta de algún proyecto, ejercicios de laboratorio y otros ejercicios asignados y/o sugeridos.**

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 (y si hay cabida) 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 y/o grupo de trabajo.** En dicha entrevista se le podrá 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 asignación le 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 en línea a través de Moodle. Luego de esa fecha en la que se entregan los resultados usted tendrá la oportunidad de hacer reclamaciones durante las **dos semanas** siguientes[3]. 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 ha recibido los mismos por razones de ausencia o desinterés.**

Laboratorios

El curso incluye una sesión de laboratorio de dos horas dirigida por un asistente de cátedra, la cual se ofrecerá 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 que nos provee el sistema Linux. 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. Recuerden que la nota que usted obtenga en esta sesión de laboratorio tendrá un peso de 10% en la nota final que va a obtener en el curso.

Importante: Entregar todos los laboratorios completados es requisito indispensable para poder aprobar el curso.

Asistencia a Clases

La asistencia a todas conferencias, así como a todos los laboratorio (según lo establece su programa de clases) es requerida por el Reglamento de la UPR. Si usted tiene que ausentarse de alguna clase por razones de peso, tendrá que mostrar evidencia en las horas de oficina. **No se acepta 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.**

Libro de Texto

El libro de texto para este curso es el siguiente:

Java Concepts (6th Edition) Cay Horstmann Wiley; 6th edition (Dec 30, 2009) ISBN-13: **978-0470509470**	

Documentos de Referencia

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:

[3]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.

- Página del texto: www.horstmann.com… **busque bajo "Java Concepts"**

  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 oficial de Java: **http://java.sun.com**
- Un tutorial de Java: **www.ibiblio.org/javafaq/javatutorial.html**
- **Java™ Platform, Standard Edition 7 API Specification**

Si busca usando "search engine" 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.

Comunicación Entre Personal del Curso y Estudiantes

Para lograr una comunicación efectiva que nos ayude a alcanzar las metas del curso estaremos utilizando varias herramientas accesibles en la Internet.

Primeramente el curso tendrán una página web donde serán publicadas todas las presentaciones, documentos y videos que utilizaremos en las conferencias. Haremos un intento por grabar todas las conferencias en forma de "screencasts" de manera que los estudiantes puedan ir nuevamente sobre alguna explicación dada en clase que no haya quedado del todo clara. La página web estará en el siguiente enlace:

http://www.ece.uprm.edu/~bvelez/courses/Fall2013/icom4015/icom4015.htm

Utilizaremos sus cuentas de correo electrónico en UPR.EDU y la cuenta de Twitter @UPRM_ICOM4015 para enviar anuncios importantes y de último momento. Es de suma importancia que verifiques tu correo electrónico frecuentemente y sigas nuestra cuenta de Twitter.

Bosquejo de Tópicos a Cubrir

En las páginas subsiguientes se presentan detalles de los tópicos a cubrir en el curso, día por día. Note que este es un detalle aproximado, pues puede ser que nos adelantemos o nos atrasemos dependiendo de la necesidad en casos particulares. El estimado de tiempo de algunos casos está basado en que ustedes ya deben tener un conocimiento de tópicos similares usando lenguajes similares a Java (lo que debe haber estudiado en el curso INGE 3016). Según vaya progresando el semestre, puede ser que, por falta de tiempo, se decida pasar por alto alguno que otro tópico entre la lista que se incluye. En ese caso, estaríamos saltando tópicos que no sean esenciales para cursos futuros.

Para cada tópico estudiado se sugieren ejercicios que se encuentran al final de los capítulos correspondientes en el texto. El propósito de estos ejercicio es que usted se mantenga practicando sobre los tópicos que se estudian en el curso, los mismos le deben dar una idea de cómo van a ser los exámenes. Con alta probabilidad, si usted hace esos ejercicios, con dedicación, no va a tener problema con los exámenes. No es que en los exámenes se vayan a preguntar los mismos ejercicios, pero **la experiencia que usted va a ganar resolviendo este tipo de ejercicio asignado es necesaria para salir bien en los exámenes. Recuerde, es necesario dedicar tiempo a sus cursos, éste no es la excepción.**

Para cada actividad, debe tomar como asignado para la lectura el material correspondiente **en el texto. Además puede sugerirse alguna lectura externa o material dado en la clase.**

Program of the Course – List of topics by Day …[4]

Days	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, and general idea of programming assignments.
1	**Chapter 1** : Introduction to Computing (most of this material should be a general review for all of you from 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: P1.4, P1.10**
2	**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. Accessing the Java API documentation. **Suggested Exercises: R2.12, R2.16, R2.17, R2.18, R2.19, P2.4, P2.10, P2.11, P2.12, P2.14, P2.15**
3	**Chapter 3: Implementing Classes.** How to implement classes in Java, implementation of simple methods, constructors, and instance and local variables. **Suggested Exercises: R3.1, R3.5, R3.6, R3.12, R3.14, R3.15, R3.20, P3.3, P3.5, P3.6, P3.7, P3.8, P3.10, P3.13, P3.15, P3.19,P3.22**
4-5	**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: R4.3, R4.4, R4.5, R4.12, R4.17, P4.13, P4.15, P4.18, P4.19**
6-7	**Chapter 5: Decisions.** We review the topic of decisions in programs, this time in the context of Java. We study selection constructs in Java, the *if-statement* and the *switch-statement*. We also study Boolean expressions and methods, and grouping of statements in blocks. **Suggested Exercises: R5.3, R5.4, R5.14, R5.15, R5.16, R5.19, R5.20, R5.23, P5.1, P5.2, P5.4, P5.5, P5.10, P5.11, P5.14, P5.18, P5.19**
8	**Exam 1**

[4]El tiempo separado para cada tópico 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 exámenes. Además, el número de periodos deja algunos periodos disponibles para exámenes, repaso, discusión de exámenes, proyectos, etc.

Day Number	Topic or Activity
9-10	**Chapter 6: Iteration.** We review the topic of iterations in programs in the context of Java. In particular, we study the *while* and for *loops* in Java. We study computer simulation as an example. **Suggested Exercises: R6.3, R6.4, R6.10, P6.2, P6.6, P6.8, P6.9, P6.10, P6.14, P6.19, P6.20**
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, R7.2, R7.3, R7.7, R7.8, R7.13, R7.15, R7.16, R7.18, P7.1, P7.4, P7.5, P7.6, P7.8, P7.9, P7.15, P7.16, P7.22**
13	**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: R8.11, R8.12, R8.13, R8.14, R8.16, R8.17, R8.20, R8.25, R8.28, P8.5, P8.13**
14	**Exam 2**
15-16	**Chapter 9: Interfaces and Polymorphisms.** Interfaces in Java, their applications in defining reusable code, classes the implement interfaces. We also study the concept of supertype and subtype, and conversions between them, polymorphisms, the use and implementation of inner classes, and the implementation of listeners for timer events. **Suggested Exercises: R9.1, R9.2, R9.3, R9.13, R9.22, R9.24, P9.6, P9.7, P9.12, P9.13, P9.19, P9.20, P9.28, P9.30, P9.35**
17-18	**Chapter 10: Inheritance.** The concept of inheritance in object oriented programming and how it is achieved in Java. **Suggested Exercises: R10.4, R10.5, R10.9, R10.10, R10.11, R10.12, P10.1, P10.2, P10.10, P10.11, P10.13, P10.16**
19	**Chapter 11: I/O & Exceptions.** We 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, R11.2, R11.8, R11.9, R11.10, R11.11, R11.12, R11.13, R11.14, R11.15, P11.7, P11.11, P11.16**
20	**Exam 3**
21	**Event Handling and Gaming.** We will study one of the most fundamental concepts in the behavior of programs with graphical user interfaces, in which the user is to have control with the proper use of the mouse, clicking buttons, filing dialog boxes, specifying next actions to execute, hence controlling what the program does. We will study what events are and how they can be controlled through a Java Program.
22-23	**Chapter 13: Recursion** **Suggested Exercises: TBA**
24-25	**Chapter 14: Sorting and Searching** **Suggested Exercises: TBA**
EXAM	**FINAL EXAM (Time and Date Set By Registrar)**