

PRONTUARIO

- A. Título del curso:** Laboratorio de Introducción a la Programación y la Ciencia de Cómputos I
- B. Codificación del curso:** COMP 3083
- C. Cantidad de horas/créditos:** Tres (3) horas contacto¹/Un (1) crédito
- D. Horas contacto:** Cuarenta y cinco (45) horas por semestre
- E. Prerrequisitos:** N/A
- F. Correquisito:** COMP 3081-Introducción a la Programación y la Ciencia de Cómputos I

G. Descripción del Curso

Este curso de laboratorio complementa la formación esperada en el curso COMP 3081 – Introducción a la Programación y la Ciencia de Cómputos I. Aquí el o la estudiante adquiere conocimiento y desarrolla destrezas para el uso de herramientas para editar, documentar, estructurar, compilar o interpretar y depurar código, tanto con interfaz textual, como de entornos gráficos de programación que integran todas estas herramientas. Se familiariza al estudiante con la documentación de interfaces de programación de aplicaciones, *API*, aprovechando, tanto de las bibliotecas del lenguaje, como las de las accesibles públicamente, para construir interfaces gráficos básicos y procesar texto. Se introduce y se ponen en práctica métodos fundamentales del desarrollo de software.

H. Objetivos de aprendizaje

Al finalizar el curso el o la estudiante podrá:

1. Enumerar los mecanismos y utilizará las herramientas necesarias para el desarrollo de software.
2. Aplicar estilos de programación y desarrollo de aplicaciones convencionales en la ciencia de cómputos.
3. Evaluar los estilos y efectos de programado escrito por otras y otros desarrolladores de su mismo nivel.

Al completar cada sección del bosquejo de contenido del curso, el o la estudiante podrá:

¹ Una hora contacto equivale a cincuenta (50) minutos.

1. Describir la función y la necesidad de programas como compiladores, intérpretes, analizadores de sintaxis y generadores de documentación que manipulan otros programas como su insumo.
2. Reconocer el vocabulario para analizar la interacción del usuario con el programado.
3. Explicar el papel de las metáforas gráficas y auditivas en la interacción del programado con el usuario y enumerar similitudes y diferencias en el uso en plataformas diversas.
4. Construir aplicaciones con una interfaz gráfica de usuario simple utilizando un estándar *API*.
5. Ejecutar procesos de diseño centrados en el usuario teniendo en cuenta la variedad de niveles de capacidad de la utilización de aplicaciones e intereses que este pueda tener.
6. Describir el modelo de programación de interfaces controlador-modelo-visualizador.
7. Escribir programas que manipulen cadenas de caracteres.
8. Describir cómo se pueden representar colores digitalmente.
9. Interpretar especificaciones, modelos y requisitos para proyectos de desarrollo de aplicaciones sencillas.
10. Ejecutar el proceso de codificación, ejecución y depuración de programas utilizando entornos de programación integrados y las herramientas que estas proveen para depurar visualmente, desarrollar pruebas unitarias y producir documentación.
11. Revisar el código enfocado en la identificación de errores comunes en la programación utilizando una guía estructurada provista para estos efectos.
12. Evaluar en qué medida el código escrito por otra persona cumple con los estilos de programación y documentación estándares y aplicarlos en su propio código.
13. Nombrar al menos un estándar para el desarrollo de interfaces.
14. Interactuar en equipo en el desarrollo de componentes de software.
15. Documentar decisiones de diseño basados en las necesidades de un grupo dado.

I. Bosquejo del contenido y distribución del tiempo

Cada tema de contenido se aborda en una sesión de tres horas continuas.

- I. Instalación de programado (3 horas)
 Programado necesario para la escritura, compilación y ejecución de código.
 Diferencias en distintas plataformas. Opciones propietarias y de código abierto. Herramientas en línea.
- II. Herramientas básicas para la edición, compilación y depuración de código (3 horas)
 Utilización de editores de texto llano, reconocimiento de fuentes y colores diferentes usados para exaltar rasgos sintácticos del código. Utilización de compilador o intérprete de comando textual, reconocimiento de informes de errores de sintaxis y localización en el código de un programa sencillo.

Reconocimiento de errores de ejecución en tales programas y utilización de técnicas rudimentarias para su localización y depuración. Localización, organización en carpetas del sistema de archivos y la utilización de nombres apropiados de los archivos generados en el proceso de programación. Empacado de trabajo para la entrega por medios electrónicos para la evaluación de parte del profesor o la profesora.

III. Ambientes de desarrollo integrado (3 horas)

Arreglo de herramientas, menús y *widgets* de los Ambientes de Programación Integrados (*IDE*) y los usos y modos de configurarlos. La noción del proyecto. Enlaces de mensajes de errores de sintaxis producidos por el *IDE* con la línea de código que puede que lo produzca.

IV. Interfaz de programación de aplicaciones (*API*) (3 horas)

Repaso de la noción de variables y sus tipos. Objetos como entidades que reciben mensajes o a las cuales se le aplican métodos. Estructura estándar de objetos de biblioteca en forma de *API*. Utilización de objetos de clases de bibliotecas.

V. Constructores de interfaces gráficas para aplicaciones (3 horas)

Widgets estándares y sus usos. Introducción al uso de un diseñador de interfaz gráfico. Inspección del código generado por el diseñador de interfaz. Archivos suplementarios generados.

VI. Programación guiada por eventos (3 horas)

Programación de interfaces gráficas sencillos que responden a eventos generados mediante la manipulación de sus *widgets*. Generación de respuesta visual al usuario.

VII. Procesamiento de texto (3 horas)

Repaso de cadenas de caracteres y las operaciones fundamentales definidas para ellas. Uso de funciones y clases de biblioteca para el pareo de patrones en cadenas de caracteres. Separación de fichas (*tokens*).

VIII. Generación de gráficas (3 horas)

Repaso de ciclos, arreglos y matrices. Representación de color. Noción de pixel. Lienzos (*canvas*) e interfaz gráfico de usuario (*GUI*) como matrices de puntos. Dibujado de elementos geométricos sencillos. Representación gráfica de datos.

IX. Herramientas de depuración (3 horas)

Repaso de trazado manual. Herramientas de depuración de sintaxis y de errores de corrida. Corrida por pasos, puntos de corte de ejecución, inspección de tablas de símbolos y valores de variables.

- | | |
|--|-------------------------|
| X. Pruebas unitarias | (3 horas) |
| Segmentos de programa o módulos que sirven para probar la corrección de funcionamiento de un módulo, función o clase (<i>unit test</i>). Herramientas de los entornos de programación integrados para apoyar esta técnica. | |
| XI. Refactorización y Abstracción procesal | (3 horas) |
| Repaso de funciones y de refinamiento de algoritmos. Estudio de programas ya escritos con el propósito de detectar segmentos de código sujetos a ser convertidos en una función (o método dentro de una clase) y aumentar así la eficiencia, claridad y reusabilidad del código. | |
| XII. Proceso de desarrollo de una aplicación | (3 horas) |
| Especificaciones para una aplicación o módulo. Ciclo de retroalimentación de especificaciones con la (el) usuario(a). Introducción a la evaluación de requerimientos. | |
| XIII. Periodos para exámenes (2). | (6 horas ²) |
| XIV. Presentación de proyecto de final de curso. | (3 horas) |

El orden de las experiencias será acoplado al orden en que se presenten los tópicos en el curso COMP 3081.

J. Técnicas Instruccionales

Con miras a lograr los objetivos del curso el o la profesora seleccionará entre las siguientes técnicas instruccionales: introducción breve de los temas, demostraciones, trabajo de programación en el laboratorio (individual o en grupo).

K. Recursos de aprendizaje e instalaciones mínimas disponibles o requeridos

La Universidad debe proveer un laboratorio para trabajo independiente de los estudiantes, el equipo electrónico que necesita el profesor o la profesora para impartir la clase, el programado apropiado para el curso (intérpretes y compiladores de todos los lenguajes que se utilizarán, herramientas para el desarrollo de aplicaciones) y acceso a la Internet. Se sugiere al estudiante poseer una computadora portátil con capacidad de ejecutar los compiladores, intérpretes u otros programados utilizados en este curso.

L. Técnicas de evaluación

Las evaluaciones consisten de: dos exámenes, trabajos de laboratorio, y un proyecto final. Se sugiere pesos relativos de 20% para exámenes, 60% para trabajos de laboratorio y 20% para el proyecto final.

² Se requieren exámenes de tres horas para poder incorporar a la evaluación ejercicios de programación que, por más sencillos que sean, requerirán al menos 30 minutos cada uno para ser completados.

M. Acomodo razonable

La Universidad de Puerto Rico (UPR) en Humacao reconoce el derecho que tienen los estudiantes con impedimentos a una educación post-secundaria inclusiva, equitativa y comparable. Conforme a la política de la Universidad de Puerto Rico hacia los estudiantes con impedimentos, fundamentada en la legislación protectora federal y local vigente en Puerto Rico, todo estudiante cualificado con impedimentos, tiene derecho a la igual participación de aquellos servicios, programas y actividades que están disponibles para la comunidad universitaria en general. Es decir, todo estudiante que tiene un impedimento de naturaleza física, mental o sensorial que lo limita sustancialmente en una o más actividades principales de la vida, podría tener derecho a recibir acomodos o modificaciones razonables en sus áreas de estudios.

Es importante que, de usted requerir algún tipo de acomodo o modificación razonable en este curso, deberá notificar al(a) profesor(a) de su necesidad de acuerdo a la condición física, mental o sensorial que presenta. Además, usted deberá visitar la Oficina de Servicios para Estudiantes con Impedimentos (OSEI) del recinto de Humacao para recibir orientación sobre los pasos que deberá seguir para solicitar acomodos o modificaciones razonables y de otros servicios de apoyo que le permitan la obtención de un adecuado aprovechamiento académico. Véase Certificación 133 (2015-2016) de la Junta de Gobierno de la UPR.

N. Integridad académica

La Universidad de Puerto Rico promueve los más altos estándares de integridad académica y científica. El Artículo 6.2 del Reglamento General de Estudiantes de la UPR (Certificación Número. 13, 2009-2010 de la Junta de Síndicos) establece que “la deshonestidad académica incluye, pero no se limita a: acciones fraudulentas, la obtención de notas o grados académicos valiéndose de falsas o fraudulentas simulaciones, copiar total o parcialmente la labor académica de otra persona, plagiar total o parcialmente el trabajo de otra persona, copiar total o parcialmente las respuestas de otra persona a las preguntas de un examen, haciendo o consiguiendo que otro tome en su nombre cualquier prueba o examen oral o escrito, así como la ayuda o facilitación para que otra persona incurra en la referida conducta”. Cualquiera de estas acciones estará sujeta a sanciones disciplinarias en conformidad con el procedimiento disciplinario establecido en el Reglamento General de Estudiantes de la UPR vigente.

O. Normativa sobre discrimen por sexo y género en modalidad de violencia sexualⁱ

“La Universidad de Puerto Rico prohíbe el discrimen por razón de sexo y género en todas sus modalidades, incluyendo el hostigamiento sexual. Según la Política Institucional contra Hostigamiento Sexual, Certificación Núm. 130 (2014-2015) de la Junta de Gobierno, si un(a) estudiante es o está siendo afectado por conductas relacionadas a hostigamiento sexual, puede acudir a la Oficina de la Procuraduría Estudiantil, el Decanato de Estudiantes o la Coordinadora de Cumplimiento con Título IX para orientación y/o para presentar una queja”.

P. Sistema de calificación

Se adjudicará la calificación A, B, C, D, o F según el nivel de competencia demostrado en las evaluaciones. El profesor o profesora podrá utilizar la escala 100-85 A, 84-75 B, 74-60 C, 59-50 D, 49-0 F, u otra que resulte más apropiada para asignar las calificaciones del curso y la informará, durante las primeras semanas de clases, y en la Guía del Estudiante.

Q. Bibliografía

- Deitel, P. J. & Deitel, H. M. (2017). *C++ how to program* (10th ed.). Upper Saddle River: Pearson Education. ISBN 978-9332585737.
- Deitel, H. M. & Deitel, P. J. (2017). *Java How to Program* (early objects) (10th ed.) Prentice Hall. ISBN 978-0134743356.
- Gaddis, T. (2017). *Starting Out with C++: From Control Structures through Objects*. (9th ed.) Addison-Wesley Publishing Company. ISBN 978-0134498379.
- Sotero Esteva, J. O. (2017). *Introducción al desarrollo de aplicaciones utilizando Scala*. Manuscrito. Universidad de Puerto Rico en Humacao: Humacao, Puerto Rico.
- Zelle, J. M. (2011). *Python programming: an introduction to computer science*. (3rd ed.) Franklin, Beedle & Associates, Inc. ISBN 978-1590282755.

R. Créditosⁱⁱ

Primera versión por Prof. José O. Sotero Esteva con el insumo de Profa. Idalyn Ríos, Prof. Jorge Montoya y Dr. Elio Ramos Colón, octubre de 2014.

Revisado por Dr. Ollantay Medina, Dr. Elio Ramos, Profa. Idalyn Ríos y Prof. José O. Sotero Esteva, 3 de enero de 2019.

Revisión por Comité Currículo del Departamento de Matemáticas, marzo de 2019.

Endosado en reunión ordinaria del Departamento de Matemáticas, 12 de abril de 2019.

Revisión general final por Profa. Bárbara L. Santiago Figueroa, abril de 2019.

Algunos de los objetivos de aprendizaje han sido adaptados de las recomendaciones contenidas en el *Information Technology Curricula 2017* de las organizaciones profesionales *Association for Computing Machinery* y el *Institute of Electrical and Electronic Engineers – Computer Society* (Ver Bibliografía).

Traducción del texto:

“The University of Puerto Rico prohibits discrimination based on sex, sexual orientation, and gender identity in any of its forms, including that of sexual harassment. According to the Institutional Policy Against Sexual Harassment at the University of Puerto Rico, Certification Num. 130, 2014-2015 from the Board of Governors, any student subjected to acts constituting sexual harassment, may turn to the Office of the Student Ombudsperson, the Office of the Dean of Students, and/or the Coordinator of the Office of Compliance with Title IX for an orientation and/or formal complaint”.

ii Este prontuario es propiedad de la Universidad de Puerto Rico. Sin embargo, su contenido es una creación académica de su autor. Por consiguiente, en el caso de uso de porciones significativas del mismo se espera reconocimiento del autor tal y como se ha hecho en esta sección.