

CARTA AL ESTUDIANTE. I Semestre, 2014
CI-0202 PRINCIPIOS DE INFORMÁTICA

Tipo de curso: Teórico – práctico.

Número de créditos: 4

Número de horas semanales presenciales: 4

Requisitos: MA-225 o equivalente (MA-230, MA-205, MA-1001)

Horario del curso: S: 8:00 – 11:50

Profesor: Dr. Juan Carlos Briceño

Correo Electrónico: juan.briceno@ecci.ucr.ac.cr

Horario de Consulta: S: 13:00 – 15:00

1. Descripción del curso:

El curso de Principios de Informática, es un curso de introducción a la programación para estudiantes del área de ingeniería y afines, donde el estudiante aprende a diseñar, construir y codificar estructuras de datos para diseñar y programar algoritmos.

2. Objetivo General:

Proveer formación básica en la implementación de programas orientados al área de ingeniería y afines.

3. Objetivos Específicos:

Al finalizar este curso el o la estudiante será capaz de:

- Diseñar algoritmos para resolver problemas específicos.
- Usar un ambiente de programación para editar, probar y depurar programas.

4. Contenidos:

Tema 1: Fundamentos de la Programación

- Arquitecturas computacionales y lenguajes de programación: lenguaje máquina, ensamblador, lenguajes de alto nivel, interpretador/compilador y paradigmas
- Ciclo de vida de un programa: problema, análisis, diseño, implementación y prueba
- Algoritmo: concepto, primitivas y ejemplos

Tema 2: Introducción a la Programación Orientada a Objetos

- Paradigma: clases e instancias, atributos y métodos, abstracción y reutilización
- Análisis y diseño: modelado de clases e instancias.
- Compilación y ejecución en lenguaje Java

Tema 3: Sistemas Numéricos y Representación de Datos

- Bases y conversión: decimal, binaria y hexadecimal
- Sistemas de codificación: ASCII y UNICODE

Tema 4: Tipos de Datos

- Tipos de datos: primitivos (enteros, reales, booleano y caracteres) e hileras
- Precisión y literales: entero (byte, short, int y long), real (float y double), booleano, carácter e hilera (secuencias de escape)

Tema 5: Definición de Variables

- Declaración: tipo, identificador y dirección
- Inicialización: tipo primitivo (valor), instancia (referencia) y estado de memoria
- Asignación y conversión (*casting*)

Utilización de Variables

- Atributos de clase: declaración, ciclo de vida y ocultamiento (encapsulamiento)
- Variables locales: declaración y ciclo de vida
- Estáticas y constantes: declaración y ciclo de vida

Tema 6: Entrada y Salida Básica

- Entrada: parámetros de línea de comandos y diálogo
- Salida: línea de comandos y diálogo

Manejo de Excepciones

- Concepto, ejemplos y definición (clase), lanzamiento y atrape

Tema 7: Expresiones y Operadores

- Aritméticos binarios (multiplicativos y aditivos) y unarios (negación y posfijos) Relacionales (comparación e igualdad), lógicos (binarios y unarios) y asignación
- Evaluación y prioridad (orden de precedencia)

Tema 8: Instrucciones y Estructuras de Control

- Estructuras de secuenciación ({})
- Estructuras de selección o bifurcación (if/else y switch)
- Estructuras de repetición o iteración (while, do y for)

Tema 9: Métodos: Fundamentos

- Conceptos: modularización y reutilización, declaración e invocación
- Componentes: encabezado (identificador, parámetros y tipo de retorno) y cuerpo
- Métodos estáticos (funciones) y ejemplos
- Sobrecarga: declaración, firmas y resolución de llamados

Métodos: Funcionamiento

- Paso de argumentos: por valor y por referencia
- Estado de memoria estática, memoria dinámica y pila de llamados
- Reglas de alcance o ámbito en identificadores

Constructores

- Concepto y utilización: declaración e invocación
- Orden de llamados

Tema 10: Recursividad

- Concepto y utilización
- Orden de llamados

Tema 11: Arreglos o Vectores: Fundamentos

- Concepto, estructura y estado de memoria
- Tipos de Datos: primitivos e instancias
- Declaración e inicialización
- Acceso a celdas y recorrido
- Parámetros de tipo arreglo y paso de argumentos

Arreglos o Vectores: Operaciones

- Utilidad y operaciones comunes (suma, promedio, mínimo, máximo)
- Búsqueda: primitivos e instancias
- Ordenamiento: primitivos e instancias
- Asociación, indexación y clasificación

Tema 12: Matrices

- Concepto, estructura y estado de memoria
- Declaración e inicialización
- Acceso a celdas y recorrido

Tema 13: Hileras o Cadenas de Caracteres

- Concepto, estado de memoria y tipos: estático y dinámico
- Operaciones: concatenación, obtener tamaño, extraer carácter o fragmento, comparación, búsqueda, reemplazo, conversión a mayúscula o minúscula, conversión a arreglo

Tema 14: Entrada y Salida: Archivos

- Conceptos y organización física de archivos
- Información y operaciones de gestión: información, creación y borrado
- Procesamiento binario/textual: apertura/cierre y lectura/escritura

Tema 15: Programación avanzada desarrollando uno de los siguientes temas:

- Matrices, algoritmos y bibliotecas de álgebra lineal.
- Fundamentos de graficación y de interfaces gráficas.
- Punteros y referencias, copia y clonación de objetos.
- Algoritmos de búsqueda y ordenamiento básicos.
- Herencia y Polimorfismo

5. Metodología:

- Los estudiantes del profesor del curso deben matricularse en el sitio:
 - <http://mediacionvirtual.ucr.ac.cr/>
 - En este sitio los estudiantes deberán subir las tareas asignadas para entrega.
 - Este sitio se conocerá con el nombre de: **el repositorio del curso.**
- El curso consistirá en clases teóricas y ejercicios ejemplo que se realizarán en el aula y clases prácticas que se realizarán en el laboratorio de cómputo donde se realizarán ejercicios de programación.
- Se realizarán pruebas cortas sin previo aviso. En estas se evaluarán ejercicios programados, prácticas de laboratorio, comprobaciones de lecturas asignadas, o de materia vista en clases anteriores.
- Los laboratorios consisten en prácticas que deben ser realizadas durante la clase de laboratorio y **subidos a su finalización en el repositorio del curso.**
- Las tareas consisten en asignaciones basadas en la materia del curso. El profesor indicará un día y hora máxima de entrega. **Las tareas deberán ser subidas en el repositorio del curso.**
- En cualquier evaluación que consista en la entrega de un programa se debe entregar el código fuente completo en el formato adecuado para poder ser revisado, compilado y ejecutado.
- Los exámenes serán realizados en el aula o en la clase de laboratorio en caso de tener una parte práctica.
- Sólo se reponen evaluaciones según lo estipulado en el artículo 24 del Reglamento de Régimen Académico de la UCR y utilizando el procedimiento allí descrito

6. Evaluación

<i>Descripción</i>	<i>Porcentaje</i>
Examen parcial 1	20%
Examen parcial 2	20%
Examen parcial 3	20%
Tareas cortas	15%
Practicas Programadas	15%
Pruebas cortas y Laboratorios	10%

Total: 100%

7. Bibliografía:

Herramientas: Sitios en Internet

- DrJava is a lightweight development environment for writing Java programs.
<http://www.drjava.org/>
- Jeliot 3 is a Program Visualization application.
<http://cs.joensuu.fi/jeliot/>
- JAMA: A Java Matrix Package.
<http://math.nist.gov/javanumerics/jama/>

Notas del curso :

- <http://mediacionvirtual.ucr.ac.cr/>

Libros de textos:

- Deitel, H.M.; Deitel, P.J. “Java Cómo programar 7ta edición”, Prentice-Hall, 2007.
www.deitel.com/books/jhtp7
www.pearsoneducacion.net/deitel
- Barnes, David J. & Kölling Michael, “Programación orientada a objetos con Java”, ISBN: 978-84-8322-350-5, Pearson Educación, 2007.
<http://www.bluej.org/objects-first/>
<http://www.bluej.org/objects-first/resources/projects.zip>
- Ceballos, Francisco Javier, “Java 2 - Curso de Programación - 3º ed.”, ISBN 970-15-1164-6, Alfaomega Ra-Ma, 2006.
http://www.fjceballos.es/publicaciones_java.html
<http://www.ra-ma.es/down/Java2-IGyApIn3ed-Ceballos.zip>
- Joyanes Aguilar, Luis. & Zahonero Martinez, Ignacio, “Programación en Java 2”, ISBN: 84-481-3290-4, McGraw-Hill, 2002.
- Schildt, Herbert, Manual de Referencia, Java 2 – 4º ed”, McGraw-Hill, 2002.