
Universidad de Costa Rica

Sede de Occidente

Ingeniería Industrial

Sigla del curso: CI-0202

Nombre del curso: Principios de Informática

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

Horas lectivas semanales: 4

Créditos: 4

Horario: V: 1:00 p.m. – 4:50 p.m.

Horas consulta: K: 2:00 p.m. - 4:30 p.m.

Profesor: MCI. Alberto Elías Ramírez Molina

Correo electrónico: alberto.ramirezmolina@ucr.ac.cr,

rammoli@gmail.com

Messenger: [rammoli@live.com](https://www.live.com)

Tel: 8816-2466

II Semestre, 2010



I Descripción del curso

El curso de Principios de Informática, es un curso básico de programación para estudiantes el área de ingeniería y afines.

En el curso se introduce al estudiante al concepto del pensamiento abstracto en la resolución de situaciones informáticas, mediante el uso ordenado y adecuado de resolución y construcción de problemas.

El estudiante aprende a diseñar, construir y reconocer la utilización de las estructuras de datos básicas de la actividad informática para lograr el diseño e implementación de algoritmos.

II Objetivo General

Proveer formación básica en programación utilizando técnicas actuales para lograr la resolución de problemas mediante la construcción de programas orientados al área de ingeniería y afines.

III Objetivos específicos

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

- Diseñar, organizar e implementar algoritmos para resolver problemas específicos
- Usar un ambiente de programación para la edición, prueba y depuración de programas
- Reutilizar componentes de software existentes en una plataforma abierta
- Aplicar a nivel básico buenas prácticas de construcción de software

IV Contenido del curso

Tema 1: Fundamentos de la Programación

- Lenguajes de programación: concepto de programación, lenguaje máquina, lenguaje ensamblador, lenguaje de alto nivel, maquina virtual, 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: modelaje de clases e instancias (UML)

-
- 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 *hileras* (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 (i.e. *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

V Metodología

- El curso consistirá en clases teóricas que se realizarán en el aula y clases prácticas que se realizarán en el laboratorio de cómputo.
- El profesor utilizará fundamentalmente la tradicional pizarra para exponer sus explicaciones y razonamientos. También se utilizará en clase el proyector multimedia y computadora, que resulta muy útil para la introducción de conceptos o la explicación de algoritmos.

- En las clases prácticas se realizarán diferentes ejercicios de programación
- Durante las clases teóricas se podrán realizar pruebas cortas o quices sin previo aviso. Dichas pruebas podrán ser de comprobación de tareas cortas, tareas programadas, prácticas de laboratorio, comprobaciones de lectura, comprobaciones de la materia vista durante la clase o en clases anteriores.
- Los laboratorios consisten en prácticas que deben ser realizadas durante la clase de laboratorio y entregados al final de la misma, a no ser que el profesor indique una fecha posterior para entregarlo. Los laboratorios también pueden ser realizados sin previo aviso y pueden ser de práctica sobre algún tema visto en una clase anterior o sobre alguna investigación corta relacionada con algún tema del curso.
- El valor de cada quiz y laboratorio depende de la cantidad de pruebas que se realicen durante el curso.
- Las tareas consisten en asignaciones basadas en la materia del curso, pueden ser de investigación. El profesor indicará un día y hora máxima de entrega. El valor de cada tarea estará indicado en el enunciado de la misma.
- 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.

VI Evaluación

Descripción	Porcentaje
Examen Parcial 1	20%
Examen Parcial 2	20%
Examen Parcial 3	20%
Tareas cortas	15%
Prácticas Programadas	15%
Quices y laboratorios	10%
T O T A L	100%

VII Bibliografía

Herramientas: Sitios en Internet

- Java SE Downloads.
<http://java.sun.com/javase/downloads/>
- Jeliot 3 is a Program Visualization application.
<http://cs.joensuu.fi/jeliot/>
- DrJava is a lightweight development environment for writing Java programs.

<http://www.DrJava.org>

- **BlueJ is an integrated Java environment specifically designed for introductory teaching.**
<http://www.bluej.org/>
- **JAMA: A Java Matrix Package.**
<http://math.nist.gov/javanumerics/jama/>
- **Chart2D is an minimalistic charting library.**
<http://jchart2d.sourceforge.net/>

Libros de textos

- **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>
- **Deitel, H.M.; Deitel, P.J. “Java Cómo programar 5ta edición”, ISBN 970-26-0518-0, Prentice-Hall, 2004.**
 - <http://www.deitel.com/>
 - ftp://ftp.prenhall.com/pub/deitel/J_HTP/java-htp5e/examples/java5-examples.zip