

Programa del curso CE-2103

Algoritmos y Estructuras de Datos II

Área Académica Ingeniería en Computadores.

[Última revisión de la plantilla: 31 de agosto del 2017]

1 Datos generales

Nombre del curso:	Algoritmos y Estructuras de Datos II
Código:	CE-2103
Tipo de curso:	Teórico – Práctico
Electivo o no:	No
Nº de créditos:	4
Nº horas de clase por semana:	4
Nº horas extra clase por semana:	12
Áreas curriculares	Ciencias de Ingeniería Diseño en Ingeniería
Ubicación en el plan de estudios:	Tercer Semestre
Requisitos:	CE-1103: Algoritmos y Estructuras de Datos I EM-1601: Matemática Discreta
Co-requisitos:	No
El curso es requisito de:	CE-3104: Lenguajes, compiladores e intérpretes EL-3307: Diseño Lógico
Asistencia:	Obligatorio
Suficiencia:	No
Posibilidad de reconocimiento:	Sí
Vigencia del programa:	II Semestre 2018

2 Descripción general

Este curso complementa el curso de Algoritmos y Estructuras de Datos I, ampliando las estructuras de datos y los algoritmos relacionados con ellas. Hace énfasis en el diseño de algoritmos y el análisis de estos, para determinar cuáles son más eficientes que otros.

Los atributos de acreditación relacionados a los graduados y a este curso son los siguientes:

- ◆ **Análisis de problemas (Inicial):** Capacidad de utilizar conocimientos apropiados y habilidades para identificar, formular, analizar y resolver

problemas complejos de ingeniería con el fin de llegar a conclusiones fundamentales.

- ◆ **Investigación (Inicial):** Capacidad para realizar investigaciones de problemas complejos por métodos que incluyen experimentos adecuados, análisis e interpretación de datos, y síntesis de la información con el fin de llegar a conclusiones válidas.
- ◆ **Trabajo en equipo (Medio):** Capacidad de trabajar efectivamente como miembro y líder de equipos, preferiblemente en un entorno multidisciplinario.

Este curso es de vital importancia para el resto de los cursos que se imparten en la carrera ya que sienta las bases de pensamiento lógico y diseño lo cual es fundamental para comprender conceptos complejos que se estudiarán más adelante.

3 Objetivos

General:

Profundizar los conocimientos relacionados con algoritmos y estructuras de datos, apoyándose en el uso de técnicas de diseño y medición de los mismos.

Objetivos específicos	Atributos correspondientes
Mejorar el diseño y modelado de estructuras de datos en aplicaciones mediante el uso de buenas prácticas.	AP(I), IN(I), TE(M)
Implementar algoritmos para la manipulación de texto.	AP(I), IN(I), TE(M)
Medir la eficiencia de algoritmos.	IN(I), TE(M)
Desarrollar diferentes estrategias para el diseño de algoritmos.	AP(I), IN(I), TE(M)
Implementar diferentes técnicas para el manejo eficiente de la memoria secundaria.	AP(I), IN(I), TE(M)
Implementar el diseño y modelado de estructuras de datos y aplicaciones.	AP(I), IN(I), TE(M)

Aprender conceptos introductorios de sistemas de archivos y sistemas de bases de datos.	IN(I)
Aplicar patrones de diseño en la solución de un problema.	AP(I), IN(I), TE(M)

4 Contenidos

→ Introducción a la administración de memoria **(2.5 semanas)**

- ◆ Referencias en memoria
- ◆ Declaración y uso eficiente de variables.
- ◆ Parámetros por referencia y por valor.
- ◆ Pila de llamadas
- ◆ Técnicas de la administración de memoria en el heap.
- ◆ Aritmética de direcciones.

→ Análisis de algoritmos **(2 semanas)**

- ◆ Medición empírica
- ◆ Medición simulada
- ◆ Medición Analítica
- ◆ Notaciones más comunes (O grade, o pequeña Ω)

→ Algoritmos de Búsqueda **(1 semana)**

- ◆ Búsqueda Secuencial
- ◆ Búsqueda Binaria
- ◆ Búsqueda por Interpolación
- ◆ Hashing
- ◆ Pathfinding

→ Patrones de Diseño, UML y calidad de código **(2 semanas)**

- ◆ Patrones constructivos
- ◆ Patrones estructurales
- ◆ Patrones de comportamiento
- ◆ Buenas prácticas de escritura de código.
- ◆ Casos de pruebas
- ◆ Tipos de pruebas
- ◆ Pruebas unitarias

→ Diseño de algoritmos (**3.5 semanas**)

- ◆ Ávidos
- ◆ Divide y vencerás
- ◆ Programación dinámica
- ◆ Backtracking
- ◆ Algoritmos probabilísticos
- ◆ Algoritmos genéticos
- ◆ Algoritmos heurísticos
- ◆ Bitvectors

→ Estructuras de almacenamiento externo (**2.5 semanas**)

- ◆ Estructuras jerárquicas (Árboles B y R)
- ◆ Sistemas de archivos
- ◆ Operaciones sobre archivos
- ◆ Archivos secuenciales
- ◆ Archivos relativos
- ◆ Archivos secuenciales indexados
- ◆ Archivos multillave
- ◆ Soluciones híbridas
- ◆ Cache de datos
- ◆ Algoritmos de ordenamiento externo

→ Algoritmos de compresión (**1.5 semanas**)

- ◆ Sin pérdida de datos
- ◆ Con pérdida de datos
- ◆ Códigos de huffman.
- ◆ Codificación aritmética.
- ◆ LZW

→ Estructuras básicas de bases de datos (**1 semana**)

- ◆ Introducción a los motores de bases de datos.
 - ◆ Definición de tablas e índices.
 - ◆ Operaciones básicas de SQL
 - ◆ Conceptos avanzados de bases de datos.
 - ◆ NOSQL
-

5 Metodología de enseñanza y aprendizaje

Se emplearán técnicas de clases magistrales por parte del profesor, en donde se desarrollarán los aspectos teóricos y prácticos más relevantes de los diferentes temas. Además se combinarán con una alta participación por parte de los estudiantes durante el transcurso de las lecciones, por medio de llamadas orales, respuestas a casos en la pizarra y de trabajos en grupo.

Se presupone que el alumno profundiza los temas abordados en la clase en las lecturas recomendadas por el profesor y que el estudiante será responsable de desarrollar los diferentes entregables que se asignen en el curso.

El profesor asumirá el papel de facilitador y el estudiante tendrá la mayor responsabilidad de su progreso.

Las herramientas básicas para poder enfrentar este curso son:

1. Una computadora con mínimo 1GB de RAM y CPU Pentium 4.
2. Sistema Operativo GNU Linux (se recomienda <http://www.ubuntu.com/>)
3. GNU g++
4. Un editor de código fuente (se recomienda C Lion)

Si un estudiante no cuenta con estos recursos puede hacer uso de los diferentes laboratorios disponibles en el campus central del Instituto Tecnológico de Costa Rica.

6 Evaluación

Pruebas cortas	10%
Tareas extra clase	10%
Tarea corta	5%
Examen final	15%
Proyectos programados	60%
TOTAL	100%

Las pruebas cortas se efectuarán en el momento de la clase que el profesor considere más apropiado. Por ningún motivo se realizarán reposiciones de pruebas cortas.

Se asignarán tareas extra que pueden estar relacionadas con análisis, diseño, programación, investigación y presentaciones. Se establecerán las reglas para cada una cuando se asignen.

Para el cálculo del porcentaje de pruebas cortas y tareas extra clase, se corresponderá a calcular la proporción de todas las calificaciones obtenidas, donde cada entregable tendrá un peso idéntico para el cálculo de dicho rubro. Para los proyectos programados no necesariamente aplicará la misma regla.

Los proyectos programados así como las tareas cortas se podrán realizar en grupos o de manera individual, si alguna de estas solo se puede realizar de forma individual se indicará en el enunciado. Las evaluaciones programadas deberán tener una adecuada documentación y los resultados deberán ser presentados al profesor en una defensa preparada adecuadamente. La defensa es obligatoria para la revisión del proyecto programado, de no ser así, no se procederá a calificar el mismo. Las normas para la documentación se entregarán oportunamente.

El examen cubrirá los contenidos del curso, y más que una comprobación de lectura o de memorización de contenidos, intentará determinar si el estudiante desarrolló los conocimientos y el criterio para aplicar los temas cubiertos en proyectos de la vida real. No existe la posibilidad de que el estudiante se exima del mismo.

En caso de que se detecte un plagio o intento de fraude en cualquier trabajo, asignación o evaluación por parte de un estudiante, se procederá a anular el mismo y se enviará una carta al expediente del estudiante.

El orden, precisión, concisión, ortografía y redacción será evaluado en todos los documentos presentados. El profesor está en la libertad de revisar o no los documentos que no cumplan con dicho rubro.

El cronograma del curso puede ser visualizado en el siguiente enlace:

https://docs.google.com/spreadsheets/d/1Y_A4o7Hn8iuSLMXo3ZEglGOmuOw8rLTQTYxe_ZqKRE/edit?usp=sharing

7 Bibliografía

Obligatoria

- Deitel y Deitel, (2012). Cómo programar en Java. 9th ed. México: Pearson Educación.
- Drozdek, Adam, (2013). Data Structures and Algorithms in Java. 4th ed. United States: Cengage Learning Asia.

Complementaria

- Weiss, M.A., "Data Structures & Algorithm. Analisis in JAVA", I edition, Addison Wesley Longman, Harlow, Inglaterra, 1999.
- Aho, A.V., Hopcroft, J.E., Ullman, J.D., "Estructuras de datos y algoritmos", versión en español de la I edición, Addison-Wesley Iberoamericana, Buenos, Aires, Argentina, 1988.
- Joyanes, Luis y Zahonero Ignacio. Estructuras de Datos: Algoritmos, abstracción y objetos. Mc Graw Hill. 1998.

8 Cuerpo docente

Ing. Isaac Ramírez, M.SI.
Egresado del instituto tecnológico de Costa Rica

Ha impartido el curso de Algoritmos y Estructuras de Datos I y II y Bases de Datos en el Área Académica Ingeniería en Computadores desde el 2014. Desde el 2008 ha ejercido en el área de computación en distintos roles como Programador, Ingeniero de Software y Supervisor de Desarrollo de Software. Actualmente se desempeña como Arquitecto de Aplicaciones en IBM.

Contacto

asistente.isaac.itcr@gmail.com
Teléfono: 83237466 (WhatsApp, Telegram)
Skype: IsaacSNK

Ing. Luis Diego Noguera Mena, MGP.
Egresado del Instituto Tecnológico de Costa Rica.
Coordinador de Algoritmos y Estructuras de Datos I y II

Ha impartido el curso de Algoritmos y Estructuras de Datos I y II en el Área Académica Ingeniería en Computadores desde el 2016. Labora para la empresa DXC, donde se ha desempeñado como Desarrollador, Líder Técnico y actualmente es el responsable de los entregables de un equipo de desarrollo de aplicaciones web.

Contacto
Email: lnoguera@tec.ac.cr
Teléfono: 88236765 (WhatsApp, Telegram)
Oficina: F2-10

Consultas

Horario y lugar de consulta: a convenir con cada profesor.

Tutorías

Horario y lugar de tutorías se indicará una vez iniciado el curso lectivo debido a que debe ser definido por la institución.

9 Instrucciones generales

Datos básicos

Complete el siguiente formulario con sus datos:

- Isaac: <https://goo.gl/forms/BUtngkylA4xaPaum1>
- Diego (Grupo 2): <https://goo.gl/forms/C0jAKCovt3XM3ov83>
- Diego (Grupo 3): <https://goo.gl/forms/UdNhwHB3gs59DTzk1>

Google Classroom

Cada estudiante deberá registrarse en el Classroom designado para el curso. A continuación los códigos de cada grupo:

- Isaac: **7aq8tm**
- Diego (Grupo 2): **2xp7mxy**
- Diego (Grupo 3): **g93yjb**

Este medio será el de preferencia para consultas, publicación de tareas, envío de entregables, citas de revisión, entre otras cosas.

Telegram

Para mensajería instantánea, se utilizará Telegram. Se tendrá un grupo de Telegram independiente para cada profesor. Cada profesor establecerá reglas de uso del grupo de Telegram.

- Isaac: No aplica
- Diego (Grupo 2): https://t.me/joinchat/AN6kkwzJKyKkYBvSf_dVng
- Diego (Grupo 3): https://t.me/joinchat/AN6kkxKnCCsFlxKFDOMg_A

Entregables

Cualquier documento que sea solicitado como entregable debe ser en formato PDF. En caso de ser necesario otro formato será indicado por el profesor.

Cada proyecto establece las reglas de entrega.

Sobre el proceso de acreditación de la carrera

Dado que la carrera Ingeniería en Computadores está en proceso de acreditación, el profesor podrá dejarse algunos de los entregables y quices, eliminando la información del estudiante.

Otros

Es terminantemente prohibido realizar grabaciones de las lecciones sin autorización del profesor.