

NOMBRE DEL CURSO: Estructuras de Datos

| | | | |
|-------------------------------|---|-------------------------------------|---|
| CÓDIGO: | 0772 | CREDITOS: | 5 |
| ESCUELA: | Ciencias y Sistemas | ÁREA A LA QUE PERTENECE: | Desarrollo de Software |
| PRE REQUISITO: | 0771 – IPC2 0796 – Lenguajes Formales y de Programación 0962 – Mate Computo 2 | POST REQUISITO: | 0722 – Teoría de Sistemas 1 0781 – Org. Lenguajes y Compiladores 2 0773 – Manejo e Implementación de Archivos |
| CATEGORIA: | Obligatorio | PERIODO: | Iro. 2019 |
| CATEDRÁTICO | Ing. Edgar René Ornélis | AUXILIAR: | José Morejón |
| EDIFICIO: | T-3 | SECCIÓN: | A |
| SALÓN DEL CURSO: | 209 | SALÓN DEL LABORATORIO: | Lab. India I, T-3 |
| HORAS POR SEMANA DEL CURSO: | 4 | HORAS POR SEMANA DEL LABORATORIO: | 2 |
| DÍAS QUE SE IMPARTE EL CURSO: | Miércoles y viernes | DÍAS QUE SE IMPARTE EL LABORATORIO: | Lunes |
| HORARIO DEL CURSO: | 07:10 – 08:50 A.M. | HORARIO DEL LABORATORIO | 09:10 – 10:50 P.M. |

DESCRIPCIÓN DEL CURSO:

El laboratorio de estructuras de datos, pretende ser el medio en donde el estudiante aplique los conocimientos obtenidos en clase, a la solución de problemas de implementación, en donde sea necesario tomar decisiones sobre la correcta y optima forma de manipular los datos, según las necesidades específicas del negocio.

Así también es la introducción a distintos frameworks de programación en el lenguaje de programación orientado a objetos Java, que en la actualidad son utilizados a nivel profesional. Estos frameworks en adelante serán de gran utilidad en la creación de aplicaciones más serias y robustas.

Al final del curso el estudiante debe presentar un nivel inicial excelente y deseos de seguir investigando y aprendiendo los frameworks de programación vistos en el laboratorio. Además de la capacidad de decidir la estructura de datos adecuada según los requisitos del contexto.

OBJETIVO GENERAL:

Que el estudiante pueda poner en práctica todos aquellos conocimientos adquiridos, y que sea capaz de identificar y aplicar de forma sistemática los distintos tipos de estructuras de datos y tecnologías para la resolución de problemas.

OBJETIVOS ESPECÍFICOS:

1. Que el estudiante tenga la habilidad de analizar e interpretar la información obtenida para que pueda formular una solución efectiva y consistente utilizando las distintas estructuras de datos.
2. Que el estudiante tenga la habilidad de construir soluciones web integrales, utilizando las distintas y tecnologías existentes.
3. Conocer y utilizar los Frameworks que formen en el estudiante un programador efectivo.
4. Que el estudiante tenga la capacidad de integrar distintas tecnologías en la elaboración de aplicaciones formales.
5. Que el estudiante sea capaz de plantear soluciones basadas en el conocimiento teórico y práctico, así como poder realizar trabajos en equipo.

HABILIDADES:

1. Toma de decisiones para utilizar estructuras de datos
2. Creatividad y originalidad en la propuesta de soluciones.
3. Efectividad al desarrollar soluciones basadas en estructuras de datos.

METODOLOGÍA:

Para el laboratorio se programaran clases presenciales, apoyadas a la vez de la elaboración de tareas prácticas para reforzar los conceptos vistos y además una parte práctica donde se resolverán problemas reales aplicando distintas estructuras y tecnologías existentes mediante la elaboración de prácticas y proyectos.

EVALUACIÓN DEL RENDIMIENTO ACADEMICO:

| Instrumento de Evaluación | Ponderación | Porcentaje |
|---------------------------|-------------|------------|
| Práctica 1 | 3.75 | 15 % |
| Proyecto 1 | 6.25 | 25 % |
| Proyecto 2 | 10 | 40 % |
| Tareas prácticas | 2.5 | 10 % |
| Final | 2.5 | 10 % |
| Total | 25 pts | 100 % |

CONTENIDO PROGRAMATICO:

Del 03 al 07

1. Introducción a las Estructuras de Datos
 - 1.1 Stack / Heap
 - 1.2 Linked List Collection
2. Herramientas, monolitos y RESTful web service
 - 2.1 Herramienta UML draw.io
 - 2.2 Servidores de aplicaciones para Frontend y Backend
 - 2.3 Monolitos
 - 2.4 RESTful web service NGREST / Javax RS
 - 2.5 Práctica 1
3. Arreglos
 - 3.1 Iteradores
 - 3.2 Búsqueda Binaria
 - 3.3 Quick Sort

Del 10 al 14

4. Árboles
 - 4.1 Integración de árboles en el framework de std::containers
 - 4.2 Recorridos con iterator/enumerators
 - 4.3 Proyecto 1
5. Tablas de dispersión
 - 5.1 Integración de tablas de dispersión en el framework de std::containers
 - 5.2 Prueba cuadrática
 - 5.3 Doble dispersión
 - 5.4 Rehashing in situ

Del 17 al 21

6. Textos
 - 6.1 Knutt Morris Pratt
 - 6.2 Boyer Moore
 - 6.3 Zip
 - 6.4 Criptografía
 - 6.5 Diccionarios

Del 26 al 28

7. Grafos
 - 7.1 Problema del bus colegial
 - 7.2 Algoritmo de Dijkstra
 - 7.3 Algoritmo de Krustal
8. Examen Final

CALENDARIZACIÓN

| Actividad | Fecha Inicio | Fecha Final |
|------------------|---------------------|--------------------|
| Tarea práctica 1 | 04/02/2019 | 11/02/2019 |
| Tarea práctica 2 | 25/02/2019 | 04/03/2019 |
| Tarea práctica 3 | 25/03/2019 | 01/04/2019 |
| Tarea práctica 4 | 01/04/2019 | 08/04/2019 |
| Práctica 1 | 11/02/2019 | 18/02/2019 |
| Proyecto 1 | 20/02/2019 | 20/03/2019 |
| Proyecto 2 | 25/03/2018 | 24/04/2019 |
| Examen final | 03/05/2018 | 03/05/2018 |

Bibliografía

Karumanchi, N. (2017). Data structures and algorithms made easy. [Hyderbad]: Careermonk Publications.

Malik, D. (2015). C++ programming. Stamford, CT: Cengage Learning.

Joyanes Aguilar, L. (2004). Algoritmos y estructuras de datos una perspectiva en c. Madrid: MacGraw-Hill.