

CARTA DESCRIPTIVA (FORMATO MODELO EDUCATIVO UACJ VISIÓN 2020)

| I. Identificadores de la asignatura | | | | |
|-------------------------------------|--|-------------------|-------------|---|
| Instituto: | IIT | Modalidad: | Presencial | |
| Departamento: | Ingeniería Eléctrica y Computación | Créditos: | 8 | |
| Materia: | Estructura de Datos | Carácter: | Obligatoria | |
| Programa: | Ingeniería en Sistemas Computacionales | Tipo: | Curso | |
| Clave: | IEC210896 | Horas: | 64 Totales | Teoría: 80% Práctica: 20% |
| Nivel: | Básico | | | |

| II. Ubicación | |
|---|----------------------------|
| Antecedentes: Programación de I | Clave: IEC981000 |
| Consecuente: Graficación | IEC321196 |

| III. Antecedentes |
|--|
| Conocimientos: Paradigmas y lenguajes de programación. Matemáticas discretas y lógica matemática. |
| Habilidades: Abstracción para la Solución de Problemas Algorítmicos. Dominio del inglés técnico. Habilidades de búsqueda, análisis y organización de información. Argumentación mediante lenguaje oral y trabajo en equipo. |

Actitudes y valores: Disponibilidad para seguir procesos. Honestidad académica, juicio crítico, responsabilidad, respeto, disposición para el aprendizaje y para el trabajo en equipo y personalidad emprendedora.

IV. Propósitos Generales

Los propósitos fundamentales del curso son:

* Que el estudiante resuelva problemas que requieran de una solución algorítmica utilizando las estructuras de datos más adecuadas para una solución más eficiente. El estudiante realizará la especificación e implementación de las estructuras de datos requeridas.

V. Compromisos formativos

Intelectual:

Discutir la utilidad de tipos de datos y estructura de datos.

Describir aplicaciones para cada estructura de datos.

Desarrollar estructuras de datos definidas por el usuario usando un lenguaje alto nivel.

Discutir el desempeño de implementaciones alternativas de las estructuras de datos.

Desarrollar aplicaciones que requieran del uso de las siguientes estructuras de datos: arreglos, cadenas, listas ligadas, pilas, colas y árboles.

Discutir las ventajas y desventajas del uso la implementación con memoria dinámica y estática de estructuras de datos.

Seleccionar la estructura de datos apropiada solucionar problemas algorítmicos.

Solucionar problemas aplicando una metodología de programación orientada a objetos utilizando como herramienta un lenguaje de programación de propósito general orientado a objetos.

Humano: Respeto al trabajo intelectual, honestidad académica y respeto a sí mismo y a los demás. Rápida adaptación a los cambios que la tecnología exige.

Social: Compromiso social y uso ético del conocimiento

Profesional: El estudiante tendrá una actitud más abierta para enfrentar y aprovechar las oportunidades que brinda la globalización del conocimiento, la revolución tecnológica y los nuevos paradigmas en el ejercicio de la profesión.

VI. Condiciones de operación

Espacio: Aula

Laboratorio: Cómputo

Mobiliario: mesa reorganizable y sillas

Población: Máximo 24

Material de uso frecuente:

- A) Proyector y computadora portátil
- B) Software requerido:
Ambiente de desarrollo integrado
- C) Conexión a Internet de alta velocidad
- D) Pizarrón, borrador y marcadores

Conditioners especiales:

No aplica

VII. Contenidos y tiempos estimados

| Temas | Contenidos | Actividades |
|---|---|---|
| Tema I Tipos de datos abstractos 6 sesiones | <p>Encuadre de la materia</p> <p>Contextualizar la importancia de la materia</p> <p>Tipos de Datos Abstractos (TDA)</p> <ul style="list-style-type: none">• Introducción a los TDA• Especificación e implementación• Utilización del paradigma orientado a objetos en TDA | <p>Presentación del curso, revisión y comentarios acerca del contenido, la evaluación y las políticas de la clase.</p> <p>Puesta en común de las expectativas de los estudiantes y de la metodología de la materia. Exploración de los conocimientos previos de los estudiantes respecto a los contenidos del curso.</p> <p>Descripción por parte del maestro de la importancia de la materia.</p> <p>El maestro introducirá al estudiante el concepto de tipo de dato abstracto. El estudiante realizará una búsqueda de información acerca del concepto y su importancia y establecerá la diferencia entre los niveles de abstracción lógico, físico y de aplicación.</p> <p>El maestro explicará al estudiante el uso del paradigma de orientación a objetos para implementar TDA.</p> <p>El estudiante realizará una especificación e implementará en un lenguaje de programación orientado a objetos un TDA (Se sugiere implementar el Tipo Abstracto Número Racional)</p> |
| Tema II Estructuras de datos básicas: Arreglo y Arreglo bidimensional 6 sesiones | <p>Tipo de dato Arreglo:</p> <ul style="list-style-type: none">• Concepto• Representación en Memoria• Manipulación y utilización del arreglo <p>Tipo de dato Arreglo</p> | <p>Para cada tipo de dato el alumno investigará el concepto y sus posibles aplicaciones.</p> <p>El estudiante realizará una especificación e implementará en un lenguaje de programación orientado a objetos un TDA que requiera el uso del arreglo y del arreglo bidimensional (Se</p> |

| | | |
|---|--|---|
| <p>Tema III Estructuras de Datos utilizando memoria estática: Pilas y Colas. 6 sesiones</p> | <p>Bidimensional</p> <ul style="list-style-type: none"> • Concepto • Representación en memoria • Manipulación y utilización del arreglo bidimensional <p>Tipo de dato Pila</p> <ul style="list-style-type: none"> • Concepto y Especificación • Implementación con arreglos • Aplicaciones <p>Tipo de dato Cola</p> <ul style="list-style-type: none"> • Concepto y especificación • Implementación como arreglo circular • Aplicaciones | <p>sugiere implementar el Tipo Abstracto Conjunto y Matriz)</p> <p>En sesión presencial el maestro junto con los alumnos construirá la especificación de cada tipo de dato. El estudiante realizará la implementación de cada tipo de dato en un lenguaje de programación orientado a objetos con asesoría del maestro. El estudiante resolverá e implementará un problema algorítmico que requiera la utilización de cada tipo de dato. Para el tipo de dato pila el estudiante resolverá el problema de verificar si los paréntesis de una expresión algebraica están balanceados y la evaluación de expresiones postfijas. Para el tipo de dato cola se implementará una aplicación que requiera el uso de colas para su solución.</p> |
| <p>Tema IV Estructuras de Datos lineales utilizando memoria dinámica 8 sesiones</p> | <p>Tipo de dato Lista simple</p> <ul style="list-style-type: none"> • Concepto y especificación • Implementación con memoria dinámica • Implementación de la pila con listas (Herencia). <p>Tipo de dato Lista circular</p> <ul style="list-style-type: none"> • Concepto y especificación • Implementación con memoria dinámica <p>Tipo de dato Lista doble</p> <ul style="list-style-type: none"> • Concepto y especificación • Implementación con memoria dinámica | <p>Para los tipos de datos de listas se realizará una aplicación en la que a través de un menú se utilicen las operaciones básicas de la lista (agregar, borrar, modificar y buscar). El tipo de dato lista se utilizará como una clase base para derivar (herencia) el tipo de dato pila para que el estudiante comprenda el concepto de mantenimiento y reutilización. Se recomienda implementar el tipo de dato pila y cola utilizando una lista para que el estudiante comprenda la separación entre los niveles de abstracción (lógico, físico y de aplicación)</p> <p>El estudiante investigará los conceptos y</p> |

| | | |
|---|--|--|
| <p>Tema V Estructuras de Datos no lineales: Árboles 6 sesiones</p> | <p>Introducción a los árboles</p> <ul style="list-style-type: none"> • Conceptos • Tipos de árboles • Representación de árboles en general <p>Árboles binarios</p> <ul style="list-style-type: none"> • Árbol de expresión • Recorridos • Implementación <p>Árboles binarios de búsqueda</p> <ul style="list-style-type: none"> • Búsqueda, inserción y borrado • Implementación | <p>los tipos de árboles. El estudiante realizará una investigación en artículos científicos de la utilización de árboles para resolver problemas complejos. El maestro explicará la representación de los árboles, así como la utilización de la recursividad para el recorrido de los árboles.</p> <p>Para cada tipo de árbol el maestro junto con los estudiantes realizará la especificación en sesión presencial. El estudiante implementará cada tipo de árbol en un lenguaje de programación orientado a objetos con asesoría del maestro.</p> |
|---|--|--|

VIII. Metodología y estrategias didácticas

Metodología Institucional:

- a) proceso de pensamiento lógico y crítico. Elaboración de prácticas e investigaciones consultando fuentes bibliográficas, hemerográficas y en Internet.
- b) Elaboración de reportes de práctica, lectura de artículos en lengua inglesa, actuales y relevantes.

Estrategias del Modelo UACJ Visión 2020 recomendadas para el curso:

- a) aproximación empírica a la realidad
- b) búsqueda, organización y recuperación de información
- c) comunicación horizontal
- d) descubrimiento
- e) ejecución-ejercitación
- f) elección, decisión
- g) evaluación
- h) experimentación
- i) extrapolación y transferencia
- j) internalización
- k) investigación
- l) meta cognitivas
- m) planeación, previsión y anticipación
- n) problematización

- o) procesos de pensamiento creativo divergente y lateral
- p) procesamiento, apropiación-construcción
- q) significación generalización
- r) trabajo colaborativo

IX. Criterios de evaluación y acreditación

a) Institucionales de acreditación:

Acreditación mínima de 80% de clases programadas

Entrega oportuna de trabajos

Pago de derechos

Calificación ordinaria mínima de 7.0

Permite examen único: no

b) Evaluación del curso

Acreditación de los temas mediante los siguientes porcentajes:

| | | |
|--------|-------|--------------------------|
| Tema 1 | 10% | Examen 100% |
| Tema 2 | 20% | Examen 50% Prácticas 50% |
| Tema 3 | 20% | Examen 50% Prácticas 50% |
| Tema 4 | 30% | Examen 50% Prácticas 50% |
| Tema 5 | 20% | Examen 50% Prácticas 50% |
| Total | 100 % | |

X. Bibliografía

Nell Dale, (2007) C++ Plus Data Structures, Jones and Bartlett Publisher, 4th edition.

Peter Smith, (2004) Applied Data Structures with C++, 1 st edition, Jones and Bartlett Publisher

Kasiviswanath, (2007) Data Structures Using C++ , Laxmi Publications LTD.

Joyanes Aguilar, L. (2006). *Programación en C++. Algoritmos, estructuras de datos y objetos*. Madrid: McGrawHill.(QA76.73C153 J69 2006).

Martínez, Roman; Quiroga, Elda. (2002). *Estructuras de Datos referencia práctica con orientación a objetos*. México: Thomson Learning QA76.9D35 M37 2002).

X. Perfil deseable del docente

Maestría o doctorado en las áreas de ciencias de la computación o tecnologías de información.

Ingeniero en Sistemas Computacionales con mínimo 3 años de experiencia.

XI. Institucionalización

Responsable del Departamento: Ing. Jesús Armando Gándara

Coordinador/a del Programa: Ing. Cynthia Vanessa Esquivel

Fecha de elaboración: Mayo 2011

Elaboró:

M. en C. Karla Olmos Sánchez

M. en C. Patricia Parroquín Amaya

M. en C. Victoria González de Moss

Dra. Leticia Ortega Maynez

Colaboró:

M. en C. Arturo R. Álvarez

Fecha de rediseño: Marzo 2013

Rediseño:

M. en C. Karla Olmos Sánchez

M. en C. Patricia Parroquin Amaya