

CARTA DESCRIPTIVA (FORMATO MODELO EDUCATIVO UACJ VISIÓN 2020)

I. Identificadores de la asignatura

Instituto:	IIT	Modalidad:	Presencial
Departamento:	Física y Matemáticas	Créditos:	8
Materia:	Aplicaciones Computacionales I	Carácter:	Obligatoria
Programas:	Ingeniería Física Ingeniería de Materiales	Tipo:	Curso
Clave:	CBE140106		
Nivel:	Introductorio		
Horas:	64 Totales	Teoría: 50%	Práctica: 50%

II. Ubicación

Antecedentes: [Clave](#)

Consecuente: [Aplicaciones Computacionales II](#)

III. Antecedentes

Conocimientos: Conocimientos rudimentarios del uso de la computadora

Habilidades: Uso de la computadora para escribir y editar archivos de texto

Actitudes y valores: Inclinación a la investigación y el estudio teórico, honestidad y respeto

IV. Propósitos Generales

Los propósitos fundamentales del curso son:

Que el estudiante adquiera la capacidad de analizar problemas, para poder describir sus soluciones en una secuencia de pasos, y que el estudiante aprenda a describir procesos en una secuencia de pasos discretos. Que el estudiante adquiera la habilidad de construir algoritmos, y concretarlos en un lenguaje de programación de alto nivel.

V. Compromisos formativos

Intelectual: El alumno podrá trabajar modelos matemáticos complejos, cuya solución requiera elaborarse computacionalmente, para la elaboración de proyectos de investigación. Así mismo estará en condiciones de ingresar a un postgrado de alto nivel, o dirigir proyectos de investigación en la industria

Humano: El alumno adquirirá confianza en sí mismo para enfrentar problemas relacionados con la programación, de ese modo fomentando que sea proactivo y propositivo

Social: El alumno comprenderá la relación entre sociedad, tecnología y la aplicación de los conceptos adquiridos en el curso.

Profesional: : El alumno debe de ser capaz de solucionar problemas que involucren la programación y su aplicación en áreas de frontera de la ciencia y la tecnología

VI. Condiciones de operación

Espacio: Laboratorio de matemáticas y cómputo

Laboratorio: Laboratorio de Computo

Mobiliario: Mesa y sillas

Población: 10 – 30

Material de uso frecuente:

- A) Proyector
- B) Computadora portátil.
- C) Computadoras de escritorio

Condiciones especiales:

Suite de compiladores de GNU, paquete de C++ o Python, suite de compiladores de Intel (deseable, optativo). Hardware requerido: Computadoras x64 multi-núcleos, sistema operativo Windows 8.1, Windows 10, Linux, MacOS o similares.

VII. Contenidos y tiempos estimados

Temas	Contenidos	Actividades
Tema 1: Introducción básica a las computadoras y la programación 4 sesiones (8 hrs.)	Arquitectura general de una computadora (RAM, disco duro, CPU, dispositivos de entrada y salida, periféricos) Editores de texto (Programa "Hello World" como primer ejemplo) Lenguajes y paradigmas de programación, algoritmos	Lectura, prácticas en el equipo de cómputo
Tema 2: Programación básica 8 sesiones (16 hrs.)	Estructura de un programa: preámbulo (cabecera, declaración de constantes y variables), cuerpo del programa Tipos de datos primitivos (enteros, punto flotante, caracteres y cadenas, booleanos, etc.) Constantes y variables Asignaciones y expresiones aritméticas (suma, resta, división, multiplicación, exponenciación, precedencia de operadores) Funciones internas básicas (sin(), cos(), tan(), sqrt(), etc.) Métodos de entrada y salida de datos Estilo de escritura de programas Técnicas de programación (diseño ascendente, pseudocódigo, diseño modular, programación estructurada)	Prácticas en el equipo de cómputo, escritura de programas simples
Tema 3: Estructuras de control, arreglos 7 sesiones (14 hrs.)	Arreglos y listas (arreglos unidimensionales, multidimensionales, listas simples, listas de listas) Control secuencial Expresiones lógicas (o booleanas) Estructuras de selección (condicionales: if, else if, else, case) Estructuras iterativas: Contadores y acumuladores Ciclos repetitivos (do o for) Ciclos Condicionales (while, repeat, do indefinido)	Prácticas en el equipo de cómputo, escritura de programas a nivel intermedio
Tema 4: Subprogramas 7 sesiones (14 hrs.)	Funciones, subrutinas, módulos Enlace de los argumentos entre programas y subprogramas Funciones recursivas	Prácticas en el equipo de cómputo, escritura de programas básicos a nivel avanzado
Tema 5: Tipos de datos avanzados y memoria dinámica 6 sesiones (12 hrs.)	Manejo dinámico de arreglos Datos de tipo derivado (o compuestos), árboles. Apuntadores	Prácticas en el equipo de cómputo, escritura de programas avanzados

VIII. Metodología y estrategias didácticas

Metodología Institucional:

- a) Elaboración de programas en editores de texto electrónicos, manejo del equipo de cómputo.
- b) Elaboración de proyectos de desarrollo de software a nivel principiante.

Estrategias del Modelo UACJ Visión 2020 recomendadas para el curso:

1. aproximación empírica a la realidad
2. búsqueda, organización y recuperación de información
3. comunicación horizontal
4. descubrimiento
5. ejecución-ejercitación
6. elección, decisión
7. evaluación
8. experimentación
9. extrapolación y transferencia
10. internalización
11. investigación
12. meta cognitivas
13. planeación, previsión y anticipación
14. problematización
15. proceso de pensamiento lógico y crítico
16. procesos de pensamiento creativo divergente y lateral
17. procesamiento, apropiación-construcción
18. significación generalización
19. trabajo colaborativo

IX. Criterios de evaluación y acreditación

a) Institucionales de acreditación:

Acreditación mínima de 80% de clases programadas
Entrega oportuna de trabajos
Calificación ordinaria mínima de 7.0
Permite examen único: No

b) Evaluación del curso

Acreditación de los temas mediante los siguientes porcentajes:

Contenido del Curso

Asistencia a clase	10 %
Prácticas	70 %
Proyecto final	20 %
Total	100 %

X. Bibliografía

1. Bjarne Stroustrup, "The C++ Programming Language (4th Edition)", Addison-Wesley ISBN 978-0321563842, 2013 (en inglés)
2. Referencia en internet en www.cplusplus.com
3. Referencia en internet en www.python.org
4. Michael Metcalf, John Reid, "Fortran 90/95 explained, 2nd Edition", Oxford University Press, 1999 (en inglés, contiene material aplicable a varios lenguajes de programación, no sólo Fortran)

X. Perfil deseable del docente

Doctorado o maestría en Ciencias Computacionales, Ingeniería, o Física con experiencia sólida o formación en programación

XI. Institucionalización

Responsable del Departamento: Mtro. Natividad Nieto Saldaña
Coordinador/a del Programa: Dr. Juan Francisco Hernández Paz
Fecha de elaboración: Agosto-Diciembre 2015
Elaboró: Dr. Gildardo Rivas Valles
Fecha de rediseño: 4 de abril 2016
Rediseño: Gildardo Rivas Valles