

PROGRAMA DE CURSO

Código	Nombre			
CC1001	Computación I			
Nombre en Inglés				
Computation I				
SCT	Unidades Docentes	Horas de Cátedra	Horas Docencia Auxiliar	Horas de Trabajo Personal
6	10	3	2	5
Requisitos			Carácter del Curso	
Ninguno			Obligatorio Plan Común	
Resultados de Aprendizaje				
<ul style="list-style-type: none"> • Desarrollar los razonamientos algorítmico y lógico como dimensiones fundamentales y básicas de la habilidad general para la resolución de problemas en el ámbito de la ingeniería y las ciencias físicas y matemáticas. • Incentivar el uso de notaciones, con sintaxis y semántica bien definidas, para expresar los procedimientos y metodologías de solución de problemas. • Promover el uso de tecnologías, y particularmente las tecnologías de información y comunicaciones, en las tareas formativas, profesionales y científicas. • Resolver problema valiéndose de notaciones, principios, técnicas, métodos y procedimientos de la disciplina de Computación. • Al final de curso los alumnos serán capaces de resolver problemas a través de problemas escritos en los lenguajes de programación MATLAB y Python. 				

Metodología Docente	Evaluación General
Clases de cátedras y clases auxiliares centradas en el trabajo de los alumnos. En cada cátedra los alumnos resuelven un problema. En cada clase auxiliar los alumnos resuelven varios problemas.	<ul style="list-style-type: none"> • Tres Controles (pruebas) parciales con un 20% de ponderación c/u y un examen global de resolución individual de problemas (ponderación 40%). • Proyectos (tareas) computaciones, individuales o grupales, que resuelvan problemas del ámbito de la ingeniería y de la ciencias físicas y matemáticas. • Calificación final: 70% controles y 30% tareas.

Unidades Temáticas

	Nombre de la Unidad	Duración en Semanas
1	Fundamentos de Programación	3 semanas (9 horas)
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
1.1 Conceptos de algoritmo, programa, lenguaje de programación (sintaxis y semántica), errores 1.2 Manejo de números: lectura/ escritura, variables, tipos, expresiones, asignación 1.3 Funciones: predefinidas (de biblioteca) y definidas por el programador. 1.4 Estructuras de control: secuencia, selección (<i>if-else</i>) y repetición (<i>while</i>) de instrucciones. 1.5 Recursión.	Resolver problemas de cálculo (cómputo) sencillo con números enteros y reales.	[1] caps 1, 2, 5, 6, 7

Número	Nombre de la Unidad	Duración en Semanas
2	Introducción a la Programación Orientada a Objetos	4 semanas (12 horas)
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
2.1 Conceptos: objeto, clase, atributos, métodos. 2.2 Uso de clases predefinidas para strings y archivos de texto 2.3 Definición y uso de clases definidas por el programador 2.4 Herencia 2.5 Uso de clases predefinidas para ventanas y sus componentes 2.6 Graficación de funciones	Resolver problemas de procesamiento de información (numérica y no numérica) que requieren facilidades especiales de representación, manipulación y comunicación amistosa con el usuario.	[1] caps 8, 14, 15, 16,17,18,19 [2] caps 9

Número	Nombre de la Unidad	Duración en Semanas
3	Listas y Tablas de Valores	3.5 semanas (10.5 horas)
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
4.1 Listas de valores: definición, indexación 4.2 Listas de 2 dimensiones ("matrices") y n-dimensiones 4.3 Diccionarios y tuplas 4.4 Algoritmos de búsqueda secuencial y binaria 4.5 Algoritmos de ordenamiento: selección-reemplazo, quixksort, etc 4.6 Búsqueda y ordenamiento de archivos secuenciales y de acceso directo	Resolver problemas que requieran la utilización de listas y tablas de valores.	[1] caps 9, 10, 11, 12

Número	Nombre de la Unidad	Duración en Semanas
4	Tipos de datos abstractos y Estructuras de Datos	2.5 semanas (7.5 horas)
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
6.1 Tipo de dato abstracto: especificación, uso, implementaciones, excepciones 6.2 Uso e implementación de Tipos de datos abstractos clásicos: Stack, Queue, Diccionario 6.3 Estructuras de datos: listas enlazadas, árboles binarios de búsqueda	Resolver problemas de mayor envergadura o complejidad que ameritan la definición y uso de tipos de datos con múltiples valores	[1] cap 13

Número	Nombre de la Unidad	Duración en Semanas
5	Introducción a la Computación Científica	2 semanas (6 horas)
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
5.1 Evaluación de polinomios y series 5.2 Cálculo de raíces: métodos de búsqueda	Resolver problemas del ámbito científico a través de	[3] Caps 11, 12 y 13

binaria, secante y Newton 5.3 Cálculo de área bajo la curva: métodos de rectángulos, trapecios y Simpson 5.4 Sistemas de ecuaciones lineales y manejo de matrices	métodos iterativos y de aproximaciones sucesivas en un lenguaje especializado (Matlab)	
Bibliografía		
<ol style="list-style-type: none"> 1. Downey, Allen. "Think Python – How to think like a Computer Scientist". Green Tea Press. 2008 2. Van Rossum, Guido. "Python Tutorial". Python Labs. 2003 3. Bielajew; "Introduction to Computers and Programming using C++ and MATLAB"; 2005 		

Vigencia desde:	2012
Elaborado por:	Juan Alvarez