

PROGRAMA DE CURSO

Código		Nombre		
CC100		Computación		
Nombre en Inglés				
Computer Science				
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				
<p>Al final del curso el alumno demuestra que resuelve problemas en el ámbito de la ingeniería y las ciencias físicas y matemáticas, utilizando razonamiento algorítmico y lógico, valiéndose de notaciones, principios, técnicas, métodos y procedimientos de la disciplina de la Computación.</p>				
Metodología Docente			Evaluación General	
<p>La metodología que se utilizará en el curso es activo - participativa con el uso de las siguientes estrategias:</p> <ul style="list-style-type: none"> - Clase Expositiva centrada en el estudiante. - Ejercicio práctico personal. 			<p>La Evaluación permitirá que los alumnos demuestren los resultados de aprendizaje alcanzadas en los distintos momentos del proceso de enseñanza, siendo estas:</p> <ul style="list-style-type: none"> • Controles (3 instancias) • Tareas computacionales individuales o grupales • Actividades en clases • Un examen <p>El examen dará cuenta de los resultados de aprendizaje general del curso. Cada una de las actividades requiere ser aprobada por separado con nota mayor o igual a 4.0.</p>	

Unidades Temáticas

Número	Nombre de la Unidad	Duración en Semanas
1	Fundamentos de Programación	3,5 semanas
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 (if-else) y repetición (while) de instrucciones. 1.5 Recursión 1.6 Operadores e instrucciones especiales.	El estudiante: 1. Resuelve problemas de cálculo (cómputo) sencillo con números enteros y reales.	<ul style="list-style-type: none"> CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo. [Barnes] caps 1, 3, 4 [Bielajew] cap 11 [Lewis] caps 2, 3 [Stein] caps 1, 2, 3, 4, 5, 6

Número	Nombre de la Unidad	Duración en Semanas
2	Programación Orientada a Objetos	2,5 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
2.1 Conceptos: objeto, clase, atributos, métodos. 2.2 Objetos y clases predefinidas 2.3 Objetos y clases definidas por el programador (tipos de datos abstractos simples). 2.4 Extensión de clases: herencia, polimorfismo, clases abstractas y genéricas. 2.5 Jerarquías de clases: predefinidas y definidas por el programador.	El estudiante: 1. Resuelve problemas procesamiento de información (numérica y no numérica) que necesitan facilidades especiales de representación y manipulación.	<ul style="list-style-type: none"> CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo. [Barnes] caps 2, 5, 7, 8, 9, 10, 11, 12 [Bielajew] cap 12 [Lewis] caps 4, 5, 7, 8 [Stein] caps 7, 8, 10, 13

Número	Nombre de la Unidad	Duración en Semanas
3	Interfaces de Usuario	2,5 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
3.1 Conceptos: interfaz, componentes, eventos 3.2 Componentes: ventanas, texto, botones, paneles, etc. 3.3 Eventos: activación, manejo (atención). 3.4 Dibujos: figuras simples (poligonos, óvalos). 3.5 Graficación de funciones 3.6 Animación simple.	El estudiante: 1. Resuelve problemas de interacción amistosa con el computador, utilizando diferentes medios de entrada y salida de información.	<ul style="list-style-type: none"> • CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo. [Barnes] caps 2, 5, 7, 8, 9, 10, 11, 12 [Lewis] cap 9 [Stein] caps 9, 15

Número	Nombre de la Unidad	Duración en Semanas
4	Listas y tablas de valores	2,5 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
4.1 Conceptos: arreglo, indexación, dimensiones 4.2 Arreglos de una dimensión ("vectores") 4.3 Arreglos de 2 dimensiones ("matrices") y n-dimensiones 4.4 Arreglos para representar objetos de múltiples valores (conjuntos, polinomios, stacks, colas de espera, diccionarios, etc.)	El estudiante: 1. Resuelve problemas que requieran el procesamiento de listas y tablas de valores.	<ul style="list-style-type: none"> • CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo. [Barnes] cap 4 [Bielajew] cap 11 [Lewis] cap 6 [Stein] cap 12

Número	Nombre de la Unidad	Duración en Semanas
5	Computación numérica	1 semana
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
5.1 Números reales: representación, precisión, errores. 5.2 Evaluación de polinomios y series. 5.3 Cálculo de raíces: métodos de búsqueda binaria, secante y Newton. 5.4 Cálculo de área bajo la curva: métodos de rectángulos, trapecios y Simpson. 5.5 Sistemas de ecuaciones	El estudiante: 1. Resuelve problemas numéricos con precisión y eficiencia.	<ul style="list-style-type: none"> CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo.

Número	Nombre de la Unidad	Duración en Semanas
6	Búsqueda y Ordenamiento de información	3 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
6.1 Algoritmos de búsqueda: secuencial, binaria, hashing. 6.2 Algoritmos de ordenamiento "cuadráticos": selección, burbuja, inserción 6.3 Ordenamiento eficientes de ordenamiento: mergesort, quicksort. 6.4 Búsqueda y ordenamiento de archivos secuenciales y de acceso directo 6.5 Estructuras de datos: listas enlazadas y árboles binarios.	Al final de la unidad, se espera que el estudiante resuelva problemas que requieran la búsqueda y/o el ordenamiento de información.	<ul style="list-style-type: none"> CapXXX Se sugiere indicar acá solo el número de la referencia y el capítulo. [Lewis] cap 6, 12

Bibliografía General
1. Barnes, Kölling; Objects First with Java; P-H; 2003 2. Bielajew; Introduction to Computers and Programming using C++ and MATLAB; 2005 3. Lewis, Loftus; Java Software Solutions: Foundations of Program Design; A-W; 2000 4. Stein; Interactive Programming in Java; MK



fcfm

FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

Vigencia desde:	Semestre Otoño 2007
Elaborado por:	J.Alvarez, N.Hitschfeld, J.Piquer, DCC
Revisado por:	ADD (septiembre 2009)