



Universidad Tecnológica Nacional
Facultad Regional Buenos Aires
Departamento de Ingeniería Electrónica

Informática I

Carrera:	Ingeniería Electrónica.	Curso:	2013
Asignatura:	Informática I	Código:	95-0452
Departamento:	Electrónica.	Clase:	Integradora anual
Área:	Técnicas Digitales	Ubicación:	1 ^{er} nivel
Bloque :	Tecnologías Básicas		
Horas semanales:	5 (cinco)		Obligatoria

Objetivos (Ord. N° 1077/05 CS) :

- Integrar en forma horizontal los conocimientos adquiridos en Álgebra, Geometría y Análisis Matemático volcando problemas (orientados a ingeniería) para ser resueltos por procedimientos informáticos.
- Optimizar a su vez el funcionamiento de INFORMATICA I como correlación académica e integración vertical con INFORMATICA II.
- Promover el hábito por la correcta presentación de informes y desarrollar la habilidad para el manejo bibliográfico

Objetivos Particulares :

Que los alumnos:

- Comprendan el funcionamiento de una computadora tanto en forma global como en sus principales partes constitutivas.
- Comprendan la relación entre las acciones realizadas por un lenguaje de programación y el intercambio de información a nivel de una máquina elemental.
- Apliquen cálculo numérico en la resolución de problemas en el contexto de procedimientos informáticos.
- Analicen problemas orientados a ingeniería por medio de un lenguaje de programación estructurado, adquiriendo un dominio fluido del mismo, que permita a Informática II asumir tal dominio y concentrarse en programación orientada a objetos..
- Comprender y manejar con fluidez las herramientas de desarrollo que componen un entorno de programación profesional: Compilador, Linker, archivador de librerías, y debugger
- Comenzar a incorporar buenas prácticas de programación, tendiendo a producir código reutilizable en el resto de las asignaturas del área Técnicas Digitales en la que se encuadra la presente asignatura.

Correlativas: (Ord. N° 1077/05 CS)

	Cursadas:	Aprobadas:
Para cursar:		



Para rendir:

Programa sintético(Ord. N° 1077/05):

- a) Estructura de una computadora. Sistemas de numeración y aritmética binaria.
- b) Diagramas de flujo.
- c) Introducción al lenguaje C.
- d) Control de flujo en C.
- e) Funciones en C.
- f) Punteros y arreglos en C.
- g) Estructuras y uniones en C. Campos de bits
- h) Manejo de archivos en C. Archivos de texto y archivos binarios.
- i) Uso del lenguaje C en aplicaciones de bajo nivel. Operaciones a nivel de bits. Puertos.

Programa analítico:

Programa analítico

Unidad Temática 1. Estructura de una Computadora

Antecedentes históricos. Definición de unidades fundamentales (bit y byte) y sus múltiplos. Definición de memoria. Capacidad de memoria. Tipos de memoria. Buses. Unidad Central del Proceso (C.P.U). Unidad Aritmética y Lógica (ALU). Unidad de Control. Contador de Programa. Dispositivos de entrada/salida (I/O). Ejecución de instrucciones. Fase de búsqueda de instrucciones. Secuencia de Instrucciones: Programa. Conceptos de Hardware y Software. Interacción entre ambos elementos.

Asimilación de los conceptos teóricos enunciados en el modelo teórico, de modo que los alumnos puedan verificarlos tangiblemente en diferentes plataformas: PC, Embeeded PC, Microcontroladores.

Bibliografía de Soporte: Essentials of Computer Organization and Architecture. Linda Null & Julia Lobur. Pennsylvania State University. Jones and Bartlett Publishers. Capítulo 1

**Unidad Temática 2. Sistemas de Numeración y Representación numérica.
Aritmética Binaria**

Introducción. Los sistemas de numeración y su evolución histórica.

Sistemas de numeración decimal, binario, octal y hexadecimal. Pasajes entre sistemas de números enteros y positivos.

Representación de Números signados: Convenio de signo y magnitud. Convenio de complemento a uno. Convenio de complemento a dos. Operaciones de adición y de sustracción utilizando el convenio de complemento a dos.

Representación de números fraccionales. Notación punto fijo y punto flotante. Precisión y truncado. Errores en notación de punto flotante. Formato IEEE 754.

Representación de caracteres: Binario Codificado Decimal (BCD), ASCII, EBCDIC, Unicode.

Códigos para Transmisión y grabación de datos: Codificación por No Retorno a Cero, Codificación por No Retorno a Cero Invertida, Modulación de Fase (Codificación Manchester), Modulación en Frecuencia, Run Length Limited Code (RLL).



Bibliografía de Soporte: Essentials of Computer Organization and Architecture. Linda Null & Julia Lobur. Pennsylvania State University. Jones and Bartlett Publishers. Capítulo 2

Unidad Temática 3. Introducción a la Programación Estructurada

Interpretación de enunciados. Ideas sobre programas y datos. Algoritmos. Estructuras Básicas de Programación. Secuencia. Selección. Iteración. Salto Incondicional. Eliminación del Salto Incondicional. Primer Paradigma de Programación; Programación Estructurada. Pseudocódigo. Implementación de algoritmos sencillos. Utilización del Modelo Top Down para desarrollo de problemas de programación paso a paso.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 2

Unidad Temática 4. El soporte del Sistema Operativo

Conceptos Básicos sobre Sistemas Operativos modernos. Objeto y Funciones principales de un sistema operativo, Tipos de sistemas Operativos. Referencias a Sistemas de uso cotidiano: Introducción a Linux. Herramientas de desarrollo: El Compilador. El Linker. Su relación con el Hardware y el Sistema Operativo.

Introducción al desarrollo de software: Programa fuente, programa objeto, programa ejecutable. Fases de compilación y vinculación de programas, en el entorno de un sistema operativo.

Bibliografía de soporte: Organización y Arquitectura de Computadores 5ta. Edición. William Stallings. Ed. Prentice Hall. Capítulo 7.

Unidad Temática 5. Introducción al Lenguaje C

Elementos del lenguaje C. Introducción a la sintaxis del lenguaje C. Primer ejemplo: Hola Mundo. Identificación de los elementos de sintaxis. Uso del compilador.

Tipos de datos, tamaño, y declaraciones. Constantes. Declaraciones. Operadores aritméticos, relacionales y lógicos. Cast. Jerarquía de operadores. Operadores de evaluación (expresiones condicionales). Operadores de Asignación. Precedencia. Preprocesador. Archivos de cabecera. Encabezador stdio.h. Entrada y salida con formato. Funciones básicas de entrada salida: scanf, printf, fprintf, getch, fgetc, fgets, getc, getchar, gets, ungetc.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 2. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 1.

Unidad Temática 6. Control de Flujo en Lenguaje C

Implementación de la Estructura de Selección: Proposición if. Proposición if-else.

Una variante de Estructura de Selección: Selección Múltiple. Proposición switch-case.

Estructura de Iteración: Ciclo while. Ciclo do-while. Ciclo for. Sentencias break y continue.

Control de Flujo con el Preprocesador: Compilación condicional. Implementación de algoritmos en lenguaje C.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 4. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 3

Unidad Temática 7. Funciones en Lenguaje C

Programación modular. Funciones que hemos utilizado hasta aquí. Definición de una función. Prototipo de una función. Argumentos. Headers.

Variables locales, globales, externas, estáticas, registros; reglas de alcance (scope). Clases de Almacenamiento. Proposición return.

Llamada por valor y por referencia.



Recursividad. Condiciones para implementar funciones recursivas. Ejemplo: Serie de Fibonacci.
Recursividad vs. Iteración.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 5. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 4.

Unidad Temática 8. Arreglos en C

Arreglos. Declaración de arreglos. Concepto de vector (arreglo de una dimensión) y de matriz (arreglo de dos dimensiones). Inicialización de los arreglos. Algoritmos de ordenamiento. Ordenamiento por más de un criterio. Algoritmos de búsqueda en arreglos. . Algoritmos de búsqueda e inserción en arreglos.

Ejemplos de uso de arreglos: desarrollo de histogramas, cálculo de media y desvío. Pasaje de arreglos como argumentos a funciones. Integración con Algebra: Desarrollo del algoritmo de Gauss Jordan. Detección bordes mediante el máximo de la derivada.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 6. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 5.

Unidad Temática 9. Punteros en C

Concepto de puntero. Concepto de dirección. Operadores unarios. Aritmética de punteros. Relación entre punteros y arreglos. Inicialización de punteros.

Implementación de llamadas a función por referencia.

Punteros a puntero. Vector de punteros. Ordenamiento de estructuras utilizando vector de punteros.

Inicialización de punteros y reserva de memoria: malloc y free.

Punteros vs. Arreglos multidimensionales.

Argumentos por línea de comandos. Punteros a función. Declaraciones complejas.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 7. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 5.

Unidad Temática 10. Estructura y uniones en C. Campos de Bits

Estructuras de datos . Bases de las Estructuras. Sintaxis: Definición de estructura. Acceso a los miembros de una estructura.

Punteros a estructuras. Arreglos de estructuras. Funciones y Estructuras. Estructuras Autoreferenciadas. Typedef.

Uniones. Definición. Acceso a los miembros de una unión. Campos de bit. Definición. Acceso a los miembros de un campo de bits. Constantes de Enumeración. Operadores lógicos. Operadores de desplazamiento. Aplicaciones relacionadas.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 10. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 6.

Unidad Temática 11. Manejo de Strings

Fundamentos de string y carácter. Librería de manejo de strings (string.h). Funciones de conversión de strings. Funciones de entrada salida de strings. Funciones de comparación de strings, Funciones de búsqueda de strings.

Bibliografía de Soporte: C/C++ Como Programar. Deitel. 4ta. Ed. Prentice Hall. Capítulo 6 y 7. El Lenguaje de Programación C. Brian Kernighan, Dennis Ritchie. Prentice Hall. Capítulo 7.



Unidad Temática 12. Manejo de archivos en C.

Concepto de Flujo de datos (streams). Creación, apertura y cierre de archivos: funciones creat, open, close. Concepto y significado de File Descriptor. Funciones para lectura y escritura de bajo nivel: read , y write. Accesos secuencial función lseek. Uso de las funciones de archivos para acceder a dispositivos de E/S acceso a E/S de alto nivel). Concepto "everything is a file". Espera del dispositivo de E/S, relación con el consumo de CPU y por ende de energía. Ejemplos de acceso al dispositivo de audio de una PC.

Bibliografía de Soporte: W. RICHARD STEVENS. Advanced Programming in the UNIX Environmet. Capítulos 3, 4, 5, 7, 8, 10, 14.

Unidad Temática 13. Aplicaciones de métodos numéricos.

Articulación con las asignaturas Álgebra y Geometría Analítica y Análisis Matemático I, desarrollando en lenguaje C aplicaciones que lleven a casos reales de aplicación los temas siguientes:

Determinación de raíces reales. Diversos Métodos de Resolución

Aproximación de funciones. Interpolación polinomial. Interpolación de Hermite.

Integración Numérica.

Método de los trapecios.

Método de Simpson.

Bibliografía de soporte: William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery. Numerical recipes in C. Cambridge University Press Numerical recipes in C.

Bibliografía:

- DEITEL & DEITEL. C/C++. **Como programar.**
- KERNIGHAM & RITCHIE. **El lenguaje de programación C.**
- LINDA NULL & JULIA LOBUR. **Essentials of Computer Organization and Architecture. Pennsylvania State University. Jones and Bartlett Publishers**
- WILLIAM STALLINGS. **Organización y Arquitectura de Computadores 5ta. Edición. Ed. Prentice Hall.**
- WILLIAM H. PRESS, SAUL A. TEUKOLSKY, WILLIAM T. VETTERLING, BRIAN P. FLANNERY. **Numerical recipes in C. Cambridge University Press.**
- W. RICHARD STEVENS. **Advanced Programming in the UNIX Environmet. Ed. Addison Wesley**