



Nombre de la materia : **Programación de computadoras**
Clave: **CI0000-T**
No. De horas /semana : **3**
Duración semanas: **16**
Total de Horas : **48**
No. De créditos : **6**
Prerrequisitos : **OC0000-L, CB0000-T (Cálculo I),
CB0100-T (Física I), CB0200-T (Álgebra Superior)**

Objetivo

El estudiante desarrollará las habilidades básicas de programación aplicando diferentes técnicas utilizando el lenguaje de programación C y las herramientas involucradas en el proceso de análisis, diseño, desarrollo, implementación y depuración de problemas típicos de diferente índole.

Contenido sintético

	Horas
1. Introducción al lenguaje C	3
2. Tipos de datos, constantes, operadores y expresiones	4
3. Entrada de datos por teclado y salida por consola	3
4. Sentencias de condición y de repetición	6
5. Programación estructurada usando funciones	6
Evaluación parcial 1	2
6. Arreglos y Apuntadores	8
7. Uso de cadenas de texto	3
8. Estructuras, Uniones y enumeraciones	3
9. Definición del proyecto de aplicación	2
10. Lectura y/o Escritura de archivos	3
Evaluación parcial 2	2
11. Directivas del precompilador	1
12. Seguimiento y entrega del proyecto de aplicación	2
Total	48

Bibliografía básica

C Programming Language (2nd Edition) by Brian W. Kernighan, Dennis M. Ritchie.

Bibliografía complementaria

Jesse Liberty. Teach Yourself C in 21 Days. 4th Edition. Sams Publishing

Harvey M. Deitel, Paul J. Deitel. C How to Program. 4th Edition.

Stephen Prata. C Primer Plus. Fifth Edition. Sams Publishing.



Metodología de enseñanza-aprendizaje

Revisión de conceptos, análisis y solución de problemas en clase:	(X)
Lectura de material fuera de clase:	(X)
Ejercicios fuera de clase (tareas):	(X)
Investigación documental:	(X)
Elaboración de reportes técnicos o proyectos:	(X)
Prácticas de laboratorio en una materia asociada:	(X)
Visitas a la industria:	()

Metodología de evaluación:

Asistencia:	(X)
Tareas:	(X)
Elaboración de reportes técnicos o proyectos:	(X)
Exámenes de Academia o Departamentales	(X)

Contenido desarrollado

1. Introducción al lenguaje C	3 hrs.
1.1. Introducción a los lenguajes de programación.	
1.2. Características del lenguaje C.	
1.3. Compilación de archivos fuente y estructura mínima de un programa escrito en C.	
1.4. Ejecución de programas compilados.	
2. Tipos de datos, constantes, operadores y expresiones	4 hrs.
2.1. Nombre de variables y tipos de datos.	
2.2. Operadores y expresiones aritméticas, lógicas, a nivel de bits y de asignación.	
2.3. Precedencia de operadores.	
2.4. Cálculo del valor máximo de un tipo, overflow, constantes con nombre.	
3. Entrada de datos por teclado y salida por consola	3 hrs
3.1. Entrada salida en C, instrucciones printf, scanf, getch, gets, uso de cursors.	
4. Sentencias de condición y de repetición	6 hrs
4.1. Condición: Sentencias if, if-else, if else if, ? y switch.	
4.2. Repetición: for, while, do-while, continue.	
4.3. Salto: break, exit.	
5. Programación estructurada con funciones	6 hrs
5.1. Introducción.	
5.2. Definición de una función.	
5.3. Estructura de una función y alcance de una variable.	
5.4. Paso de argumentos a una función y tipo de dato devuelto.	
5.5. Diseño top-down.	
5.6. Recursividad.	
Evaluación	2 hrs.



6. Arreglos y Punteros	8 hrs.
6.1. Introducción.	
6.2. Direcciones de variables.	
6.3. Punteros como argumentos.	
6.4. Paso por valor y por referencia en funciones.	
6.5. Aritmética de apuntadores.	
6.6. Definición de arreglo y declaración tipo vector y tipo matriz.	
6.7. Manipulación de arreglos y punteros.	
6.8. Diferencia entre variables tipo apuntador y variables tipo arreglo.	
7. Uso de cadenas de texto	3 hrs
7.1. Las cadenas como arreglos.	
7.2. Implementación de algunas funciones para manejar cadenas.	
7.3. Uso de las funciones proporcionadas por las librerías.	
8. Estructuras, Uniones y enumeraciones	3 hrs
8.1. Definición de estructura.	
8.2. Arreglos de estructuras.	
8.3. Inicialización de una estructura.	
8.4. Punteros a estructuras y punteros a arreglo de estructuras.	
8.5. Paso de estructuras a funciones o sólo miembros de ella.	
8.6. Estructuras dentro de estructuras.	
8.7. Definición de unión y diferencia con una estructura.	
9. Definición del proyecto de aplicación	2 hrs.
10. Lectura y/o Escritura de archivos	3 hrs.
10.1. Introducción a los archivos.	
10.2. Lectura: uso de FILE*, fopen, comprobar si está abierto, getc, feof, fclose, fgets, fread, fscanf.	
10.3. Escritura: putc, fputs, fwrite, fprintf.	
Evaluación	2 hrs.
11. Directivas del precompilador	1 hrs.
12. Seguimiento y entrega del proyecto de aplicación	2 hrs.

Historia de revisiones

12 de agosto del 2009. Este programa fue revisado por los profesores: M.C. Jaime Cerda Jacobo, M.I. Moisés García Villanueva, M.I. José Rafael Rodríguez Ochoa.

Mayo de 2017. Este programa rescata un acuerdo previo de la Academia de Computación para modificar los prerrequisitos indicados en amarillo. La revisión del contenido se encuentra en proceso por los profesores involucrados. Han asistido a



reuniones: Ismael Chávez, Félix Jiménez, José Rafael Rodríguez, Miguel Angel García, Violeta Medina y Leonardo Romero.

Fecha de autorización por el H. Consejo Técnico: dd/mm/aaaa

Comentarios para la Academia y el H. Consejo Técnico:

- Se modifican los prerrequisitos de esta materia, con el fin de tener alumnos más maduros, sin la preocupación de reprobar por segunda vez alguna materia de las básicas de primer semestre (álgebra superior, cálculo I o física I).
- Al haber aprobado los cursos básicos mencionados, se enriquece el abanico de problemas para abordar en la programación.
- Esta materia tiene un altísimo índice de reprobación y se ofrecen numerosos grupos en cada ocasión. Se espera que disminuya el número de grupos y disminuya significativamente el índice de reprobación.