



**PLAN DE ESTUDIOS (PE): Licenciatura en Ciencias de la Computación /
Ingeniería en Ciencias de la Computación**

AREA: Ciencias de la Computación / Ingeniería en Ciencias de la Computación

ASIGNATURA: Programación I

CÓDIGO: CCOS-004

CRÉDITOS: 6

FECHA: 25-Abril-2016





1. DATOS GENERALES

Nivel Educativo:	Licenciatura
Nombre del Plan de Estudios:	Licenciatura en Ciencias de la Computación / Ingeniería en Ciencias de la Computación
Modalidad Académica:	Presencial
Nombre de la Asignatura:	Programación I
Ubicación:	Básico
Correlación:	
Asignaturas Precedentes:	Metodología de la Programación
Asignaturas Consecuentes:	Programación II / Programación II y Ensamblador

2. CARGA HORARIA DEL ESTUDIANTE

Concepto	Horas por semana		Total de horas por periodo	Total de créditos por periodo
	Teoría	Práctica		
Horas teoría y práctica (16 horas = 1 crédito)	3	2	90	6

3. REVISIONES Y ACTUALIZACIONES

Autores:	Beatriz Beltrán Martínez Graciano Cruz Almanza Hilda Castillo Zacatelco José Luis Meza León Judith Pérez Marcial Laura Cuayahuitl Romero	Marco Antonio Soriano Ulloa Meliza Contreras González Miguel Rodríguez Hernández Pedro Bello López Yalú Galicia Hernández Yolanda Moyao Martínez
Fecha de diseño:	11 de junio de 2009	





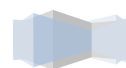
Fecha de la última actualización:	25 de abril de 2016	
Fecha de aprobación por parte de la academia de área	25 de abril de 2016	
Revisores:	Ana Patricia Cervantes Márquez Beatriz Beltrán Martínez Darnes Vilariño Ayala Erika Bonfil Barragán Eugenia Erica Vera Cervantes Guillermina Sánchez Román Hilda Castillo Zacatelco	Marco Antonio Soriano Ulloa Mario Rossainz López Meliza Contreras González Miguel Rodríguez Hernández Mireya Tovar Vidal Pedro Bello López Rafael De la Rosa Flores
	José Andrés Vázquez Flores Leticia Mendoza Alonso	Yalú Galicia Hernández Yolanda Moyao Martínez
Sinopsis de la revisión y/o actualización:	Se decidió agregar el punto 3.1.2.1 Lista ligada simple, dentro de la unidad 3, para reforzar el concepto de apuntador con una aplicación. Se realizó el cambio de formato y se pasó a competencias. La unidad 1: conceptos básicos de la computación, se movió a metodología de la programación, debido a que son conceptos que se requieren para Ensamblador y deben ser estudiados previamente.	

4. PERFIL DESEABLE DEL PROFESOR (A) PARA IMPARTIR LA ASIGNATURA:

Disciplina profesional:	Ciencias de la Computación
Nivel académico:	Maestría
Experiencia docente:	Mínima de 2 años
Experiencia profesional:	Mínima de 1 año

5. PROPÓSITO:

El estudiante obtendrá los conocimientos y habilidades para resolver problemas utilizando programación estructurada a partir de algoritmos para la solución de problemas, a través del trabajo en equipo de manera colaborativa, responsable, honesta y comprometido con su aprendizaje.





El estudiante será capaz de:

- Identificar las partes que conforman un programa, así como utilizar correctamente las diferentes estructuras de control del lenguaje.
- Analizar y aplicar las operaciones sobre arreglos y cadenas para la solución de problemas.
- Utilizar estructuras básicas para el acceso a la información de forma estática y dinámica y reconocer la importancia del uso de funciones para la solución de problemas.
- Identificar los tipos de archivos y aplicar sus operaciones para la solución de problemas.

6. COMPETENCIAS PROFESIONALES:

Licenciatura:

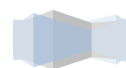
Desarrollar de forma efectiva y eficiente los algoritmos y programas apropiados para resolver problemas complejos de computación. Ingeniería:

Diseñar soluciones creativas e innovadoras por medio del análisis, síntesis e implementación en sistemas de cómputo que cumplan con los estándares de calidad.

En la materia se diseñan y desarrollan programas mediante el uso de un lenguaje de alto nivel para la resolución de problemas a través del paradigma estructurado.

7. CONTENIDOS TEMÁTICOS

Unidad de Aprendizaje	Contenido Temático	Referencias
1. Introducción al Lenguaje	1.1 Historia y desarrollo del lenguaje	Kernighan B. & Ritchie D. (1995). Lenguaje de Programación C (2ª. ed). México: Pearson Editorial
	1.2 Identificadores estándar	
	1.3 Palabras reservadas	
	1.4 Estructura de un programa	Joyanes A. & Zahonero I. (2005). Programación en C (2ª ed). Mc Graw Hill
	1.5 Tipos de datos estándar (primitivos)	Deitel, H. & Deitel, P. (2016) C, How to program, (8th ed). USA: Pearson Education
	1.6 Declaración de constantes y variables	
	1.7 Entrada y salida básica	





1.8 Estructuras de control 1.8.1 Secuencia 1.8.2 Decisión 1.8.3 Iteración
--

Unidad de Aprendizaje	Contenido Temático	Referencias
2. Arreglos y Estructuras	2.1 Arreglos unidimensionales	Kernighan B. & Ritchie D. (1995). Lenguaje de Programación C (2ª. ed). México: Pearson Editorial Joyanes A. & Zahonero I. (2005). Programación en C (2ª ed). Mc Graw Hill Deitel, H. & Deitel, P. (2016) C, How to program, (8th ed). USA: Pearson Education
	2.2 Aplicaciones sobre arreglos unidimensionales	
	2.2.1 Ordenamiento Burbuja Selección Inserción	
	2.2.2 Búsqueda Secuencial Binaria	
	2.3 Manejo de cadenas	
2.4 Arreglos bidimensionales		
2.5 Estructuras		

Unidad de Aprendizaje	Contenido Temático	Referencias
3. Apuntadores y Funciones	3.1 Definición de apuntador 3.1.1 Operaciones básicas con apuntadores y con arreglos 3.1.2 Manejo dinámico de memoria 3.1.2.1 Lista ligada simple	Kernighan B. & Ritchie (1995). Lenguaje de Programación C (2ª. ed). México: Pearson Editorial Joyanes A. & Zahonero I. (2005). Programación en C (2ª ed). Mc Graw Hill Deitel, H. & Deitel, P. (2016) C, How to program, (8th ed). USA: Pearson Education
	3.2 Funciones 3.2.1 Paso de parámetros (por valor, por referencia) 3.2.2 Alcance de una variable	





Unidad de Aprendizaje	Contenido Temático	Referencias
4. Archivos	4.1 Introducción	Kernighan B. & Ritchie (1995). Lenguaje de Programación C (2ª ed). México: Pearson Editorial Joyanes A. & Zahonero I. (2005). Programación en C (2ª ed). Mc Graw Hill Deitel, H. & Deitel, P. (2016) C, How to program, (8th ed). USA: Pearson Education
	4.2 Archivos de Texto 4.2.1 Operaciones sobre archivos de texto (apertura, lectura, escritura, cerrar archivo)	
	4.3. Archivos Estructurados 4.3.1 Operaciones sobre archivos estructurados (apertura, lectura, escritura, cerrar archivo)	

8. ESTRATEGIAS, TÉCNICAS Y RECURSOS DIDÁCTICOS

Estrategias y técnicas didácticas	Recursos didácticos
<p>Estrategias de aprendizaje:</p> <ul style="list-style-type: none"> • Lectura y comprensión, □ Reflexión, • Comparación. <p>Estrategias de enseñanza:</p> <ul style="list-style-type: none"> • Aprendizaje activo, • Aprendizaje cooperativo, • Aprendizaje colaborativo, <p>Ambientes de aprendizaje:</p> <ul style="list-style-type: none"> • Aula, • Laboratorio, • Simuladores. 	<p>Materiales:</p> <ul style="list-style-type: none"> • Proyectores, • TICs, • Plumón y pizarrón, • Ejercicios • Libros, fotocopias • Artículos científicos • Antologías • Materiales audiovisuales • Programas informáticos (CD u on-line) educativos. • Aplicaciones Multimedia • Páginas Web • Webquest • Correo electrónico
Estrategias y técnicas didácticas	Recursos didácticos





<p>Actividades y experiencias de aprendizaje:</p> <ul style="list-style-type: none"> Asistencia a congresos <p>Técnicas</p> <ul style="list-style-type: none"> Grupales, De problemas, De estudio de casos, Mapas conceptuales, Para el análisis, Comparación, Síntesis, Mapas mentales, Lluvia de ideas, <input type="checkbox"/> Portafolio, Exposición. 	<ul style="list-style-type: none"> <input type="checkbox"/> Chats <input type="checkbox"/> Foros <input type="checkbox"/> Links
---	--

9. EJES TRANSVERSALES

Eje (s) transversales	Contribución con la asignatura
Formación Humana y Social	Promover la ética en el desarrollo de programas como un elemento de formación integral, así como el interés por el cuidado del medio ambiente.
Desarrollo de Habilidades en el uso de las Tecnologías de la Información y la Comunicación	Desarrollo de aplicaciones mediante la programación de algoritmos.
Desarrollo de Habilidades del Pensamiento Complejo	Desarrollo de la habilidad de programación estructurada.
Lengua Extranjera	Lectura de artículos especializados en el área de la programación estructurada, para favorecer la comprensión de la sintaxis de los lenguajes de programación.
Innovación y Talento Universitario	Planteamiento de programas usando técnicas de resolución de problemas que puedan ser utilizados en otras disciplinas de la ciencia.
Educación para la Investigación	Fomentar el hábito de investigar diversas técnicas de programación y compararlas con algunas otras.





10. CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
<input type="checkbox"/> Tareas	10 %
<input type="checkbox"/> Prácticas de laboratorio	15 %
<input type="checkbox"/> Proyecto final	20 %
<input type="checkbox"/> Pruebas objetivas	30 %
<input type="checkbox"/> Participación en clase	10 %
<input type="checkbox"/> Asistencia	5 %
<input type="checkbox"/> Entregas puntuales	5 %
<input type="checkbox"/> Presentación de trabajos	5 %
Total	100%

11. REQUISITOS DE ACREDITACIÓN

Estar inscrito como alumno en la Unidad Académica en la BUAP
Asistir como mínimo al 80% de las sesiones para tener derecho a exentar por evaluación continua y/o presentar el examen final en ordinario o extraordinario
La calificación mínima para considerar un curso acreditado será de 6
Cumplir con las actividades académicas y cargas de estudio asignadas que señale el PE

Notas:

- a) La entrega del programa de asignatura con sus respectivas actas de aprobación, deberá realizarse en formato electrónico, vía oficio emitido por la Dirección o Secretaría Académica a la Dirección General de Educación Superior.
- b) La planeación didáctica deberá ser entregada a la coordinación de la licenciatura en los tiempos y formas acordados por la Unidad Académica.

