

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL

PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Arquitectura y Diseño, Ensenada; Facultad Ciencias Químicas e Ingeniería, Tijuana; y Facultad de Ciencias de la Ingeniería y Tecnología, Valle de las Palmas
- 2. Programa Educativo:** Ingeniero en Software y Tecnologías Emergentes
- 3. Plan de Estudios:** 2022-1
- 4. Nombre de la Unidad de Aprendizaje:** Lenguaje C
- 5. Clave:** 40030
- 6. HC:** 01 **HT:** 02 **HL:** 02 **HPC:** 00 **HCL:** 00 **HE:** 01 **CR:** 06
- 7. Etapa de Formación a la que Pertenece:** Básica
- 8. Carácter de la Unidad de Aprendizaje:** Optativa
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



Equipo de diseño de PUA

Leocundo Aguilar Noriega
Elitania Jiménez García
María Berenice Fong Mata
Mauricio Alonso Sánchez Herrera
Sergio Omar Infante Prieto

Fecha: 20 de febrero de 2021

Vo.Bo. de subdirector(es) de Unidad(es) Académica(s)

Humberto Cervantes De Ávila
Daniela Mercedes Martínez Platas
Noemí Hernández Hernández

II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

La unidad de aprendizaje brinda los conocimientos sobre las características y funcionalidades del lenguaje C, lo que permite al alumno desarrollar programas eficientes que resuelven problemas de procesamiento de información en los ámbitos de la industria e investigación, utilizando la sintaxis y biblioteca estándar del lenguaje C.

Se ubica en la etapa básica, con carácter de optativa y forma parte del área de Métodos y Tecnologías de Software del programa educativo Ingeniero en Software y Tecnologías Emergentes.

III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE

Desarrollar programas computacionales, tomando en cuenta las características y funcionalidades del lenguaje C de manera eficiente, para solucionar problemas procesamiento de información en los ámbitos de la industria e investigación, con actitud propositiva y creativa.

IV. EVIDENCIA(S) DE APRENDIZAJE

Elabora y entrega portafolio de evidencias; debe integrar el código implementado en el desarrollo y las soluciones a problemas resueltos durante taller y laboratorio. Se entrega en formato electrónico.

V. DESARROLLO POR UNIDADES
UNIDAD I. Conceptos generales del lenguaje C

Competencia:

Distinguir entre los distintos conceptos del lenguaje C, para familiarizarse con el mismo, mediante la identificación de sus principales características, con actitud analítica y reflexiva.

Contenido:

Duración: 1 hora

- 1.1. Historia del Lenguaje C
- 1.2. Variables, expresiones aritméticas y constantes simbólicas
- 1.3. Propositiones del lenguaje (condicionales y ciclos)
- 1.4. Biblioteca estándar
- 1.5. Arreglos
- 1.6. Funciones, argumentos y llamadas por valor
- 1.7. Descripción del proceso de compilación

UNIDAD II. Tipos, operadores y expresiones

Competencia:

Distinguir entre los distintos tipos de variables, operadores y expresiones del lenguaje C, mediante la identificación de sus principales características, para implementarlos de forma adecuada durante la programación, con actitud analítica y reflexiva.

Contenido:

Duración: 1 hora

- 2.1. Nombres de variables
- 2.2. Tipos y tamaños de datos
- 2.3. Constantes
- 2.4. Declaraciones
- 2.5. Operadores aritméticos, de relación y lógicos
- 2.6. Operadores de incremento y decremento
- 2.7. Operadores para manejo de bits
- 2.8. Operadores de asignación y expresiones
- 2.9. Expresiones condicionales
- 2.10. Precedencia y orden de evaluación

UNIDAD III. Control de flujo

Competencia:

Diferenciar entre las distintas estructuras de control de programa, mediante la comprensión del flujo de instrucciones que se logra con cada una de ellas, para su uso de forma adecuada durante la programación, con actitud analítica y reflexiva.

Contenido:

- 3.1. Propositiones y bloques
- 3.2. If-else y else-if
- 3.3. Switch
- 3.4. Ciclos for, while y do-while
- 3.5. Break y continue
- 3.6. Goto y etiquetas

Duración: 2 horas

UNIDAD IV. Funciones y la estructura del programa

Competencia:

Manejar la estructura general del programa, mediante la comprensión de las funciones y elementos relacionados, para su uso de forma adecuada durante la programación, con actitud analítica y lógica

Contenido:

- 4.1. Conceptos básicos de funciones
- 4.2. Funciones con retorno de valor
- 4.3. Variables y reglas de alcance
- 4.4. Variables estáticas y de tipo registro
- 4.5. Inicialización de variables

Duración: 2 horas

UNIDAD V. Apuntadores y arreglos

Competencia:

Organizar el uso de la memoria en un programa, mediante la implementación de apuntadores y arreglos, para mejorar la eficiencia de ejecución, con actitud analítica y sistemática.

Contenido:

Duración: 3 horas

- 5.1. Apuntadores y argumentos de funciones
- 5.2. Apuntadores y arreglos
- 5.3. Aritmética de direcciones
- 5.4. Apuntadores a caracteres y funciones
- 5.5. Arreglos de apuntadores y apuntadores a apuntadores
- 5.6. Arreglos multidimensionales
- 5.7. Inicialización de arreglos de apuntadores
- 5.8. Apuntadores vs. arreglos multidimensionales
- 5.9. Argumentos en la línea de comandos
- 5.10. Apuntadores a funciones
- 5.11. Declaraciones avanzadas

UNIDAD VI. Estructuras

Competencia:

Jerarquizar el manejo de datos estáticos y dinámicos, mediante su agrupación en estructuras, para facilitar su procesamiento, de forma organizada y actitud creativa.

Contenido:

Duración: 4 horas

- 6.1. Conceptos básicos sobre estructuras
- 6.2. Estructuras y funciones
- 6.3. Arreglos de estructuras
- 6.4. Apuntadores a estructuras
- 6.5. Estructuras autorreferenciadas
- 6.6. Asignación Dinámica de memoria
- 6.7. Typedef
- 6.8 Uniones
- 6.9. Campos de bits

UNIDAD VII. Archivos

Competencia:

Construir programas que almacenen y recuperen información, mediante el uso de archivos, para mantener los datos organizados y persistentes en memoria secundaria, con actitud analítica y metódica.

Contenido:

- 7.1. Introducción a archivos y flujos
- 7.2. Archivos de acceso secuencial (texto)
- 7.3. Archivos de acceso aleatorio (binarios)
- 7.4. Ejemplos de uso

Duración: 3 horas

VI. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

No.	Nombre de la Práctica	Procedimiento	Recursos de Apoyo	Duración
UNIDAD I				
1	Conceptos básicos del lenguaje C	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Analiza problemas, determinando las entradas, procesos y salidas. 3. Se documenta el análisis realizado. 4. Entrega reporte de la actividad al docente para su retroalimentación. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas
UNIDAD II				
2	Análisis de expresiones aritméticas, lógicas y relacionales	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Realiza pruebas de escritorio de un fragmento de código e interpreta su salida en pantalla. 3. Se documenta el análisis realizado. 4. Entrega reporte de la actividad al docente para su retroalimentación. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz ● Calculadora 	2 horas
UNIDAD III				
3	Comprendiendo el uso de estructuras de control de selección	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Realiza pruebas de escritorio de un fragmento de código e interpreta su salida en pantalla. 3. Se documenta el análisis realizado. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas

		4. Entrega reporte de la actividad al docente para su retroalimentación.		
4	Comprendiendo el uso de estructuras repetitivas	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Realiza pruebas de escritorio de un fragmento de código e interpreta su salida en pantalla. 3. Se documenta el análisis realizado. 4. Entrega reporte de la actividad al docente para su retroalimentación. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas
UNIDAD IV				
5	Conceptos básicos de funciones	<ol style="list-style-type: none"> 1. Atiende las indicaciones del docente para para elaborar la actividad. 2. analiza las características y ventajas de programar haciendo uso de funciones 3. Se documenta el análisis realizado, incluyendo opinión personal del tema. 4. Entrega reporte de actividad al docente para su evaluación y retroalimentación. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas
6	Funciones que retornan valores	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. investiga el comportamiento y sintaxis de las funciones que retornan valores 3. Se documenta el análisis realizado, incluyendo opinión personal del tema 4. Entrega reporte de actividad al 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas

		docente para su evaluación y retroalimentación.		
7	Variables: inicialización, variables estáticas, variables tipo registro y alcance de las variables.	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. investiga acerca de variables: inicialización, estáticas, tipo registro y alcance de las variables 3. realiza un cuadro sinóptico de los temas anteriores. . 4. Entrega cuadro sinóptico al docente para su evaluación y retroalimentación. 	<ul style="list-style-type: none"> ● Apuntes de clase ● Fuentes bibliográficas ● Lápiz 	2 horas
UNIDAD V				
8	Conceptos básicos de apuntadores	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Resuelve ejercicios provenientes de la bibliografía del curso sobre conceptos básicos de apuntadores 3. Resuelve ejercicios de problemas reales propuestos por el docente. 4. Contrasta soluciones con las de sus compañeros. 5. Entrega reporte de la actividad al docente para su retroalimentación. 	<ul style="list-style-type: none"> ● Bibliografía del curso ● Cuaderno y lápiz 	2 horas
9	Arreglos con apuntadores	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Resuelve ejercicios sobre el uso de arreglos con apuntadores provenientes de la bibliografía del curso. 	<ul style="list-style-type: none"> ● Bibliografía del curso ● Cuaderno y lápiz 	2 horas

		<ol style="list-style-type: none"> 3. Resuelve ejercicios de problemas reales propuestos por el docente. 4. Contrasta soluciones con las de sus compañeros. 5. Entrega reporte de la actividad al docente para su retroalimentación. 		
10	Conceptos avanzados de apuntables	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la actividad. 2. Resuelve ejercicios sobre los conceptos avanzados de apuntables provenientes de la bibliografía del curso. 3. Resuelve ejercicios de problemas reales propuestos por el docente. 4. Contrasta soluciones con las de sus compañeros. 5. Entrega reporte de la actividad al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Bibliografía del curso • Cuaderno y lápiz 	2 horas
UNIDAD VI				
11	Estructuras	<ol style="list-style-type: none"> 1. Resuelve ejercicios de estructuras, arreglos y apuntables a estructuras provenientes de la bibliografía del curso. 2. Resuelve ejercicios reales propuestos por el docente que requieren estructuras, arreglos y apuntables a estructuras. 3. Entrega reporte de la actividad realizada para su 	<ul style="list-style-type: none"> • Apuntes de clase • Bibliografía del curso • Cuaderno y lápiz 	2 horas

		retroalimentación.		
12	Memoria Dinámica	<ol style="list-style-type: none"> 1. Resuelve ejercicios que hacen uso de memoria dinámica provenientes de la bibliografía del curso. 2. Resuelve ejercicios de problemas reales propuestos por el docente que requieren uso de memoria dinámica. 3. Entrega reporte de la actividad realizada para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Bibliografía del curso • Cuaderno y lápiz 	2 horas
13	Uniones y Campos de bits	<ol style="list-style-type: none"> 1. Resuelve ejercicios que hacen uso de uniones y campos de bits provenientes de la bibliografía del curso. 2. Resuelve ejercicios de problemas reales propuestos por el docente que requieren uso de uniones y campos de bits 3. Entrega reporte de la actividad realizada para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Bibliografía del curso • Cuaderno y lápiz 	2 horas
UNIDAD VII				
14	Archivos de texto	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor con respecto al uso de archivos y el desarrollo de los ejercicios. 2. Investiga los principales conceptos y funciones para el manejo de archivos de texto 3. Elabora un cuadro sinóptico con los conceptos y 	<ul style="list-style-type: none"> • Bibliografía del curso • Cuaderno y lápiz 	2 horas

		<p>documentación de funciones.</p> <ol style="list-style-type: none"> 4. Resuelve ejercicios de problemas reales propuestos por el docente, en los cuales sea necesario utilizar archivos secuenciales de texto. 5. Escribe un reporte de conclusiones, estableciendo las ventajas y desventajas en el uso de archivos de texto. 		
15	Archivos de acceso aleatorio	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor con respecto al uso de archivos y el desarrollo de los ejercicios. 2. Investiga los principales conceptos y funciones para el manejo de archivos binarios de acceso aleatorio. 3. Elabora un cuadro sinóptico con los conceptos y documentación de funciones. 4. Resuelve ejercicios de problemas reales propuestos por el docente, en los cuales sea necesario utilizar archivos binarios de acceso aleatorio. 5. Escribe un reporte de conclusiones, estableciendo las ventajas y desventajas en el uso de acceso aleatorio. 	<ul style="list-style-type: none"> ● Bibliografía del curso ● Cuaderno y lápiz 	4 horas

VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No.	Nombre de la Práctica	Procedimiento	Recursos de Apoyo	Duración
UNIDAD I				
1	Conceptos básicos del lenguaje C	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la práctica. 2. Realiza programas que resuelvan los problemas planteados por el docente. 3. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Fuentes bibliográficas • Computadora con Compilador de lenguaje C • Manual de Prácticas 	2 horas
UNIDAD II				
2	Análisis de expresiones aritméticas, lógicas y relacionales	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la práctica. 2. Realiza programas que resuelvan los problemas planteados por el docente. 3. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Fuentes bibliográficas • Computadora con Compilador de lenguaje C • Manual de Prácticas 	2 horas
UNIDAD III				
3	Comprendiendo el uso de estructuras de control de selección	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la práctica. 2. Realiza programas que resuelvan los problemas planteados por el docente. 3. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Fuentes bibliográficas • Computadora con Compilador de lenguaje C • Manual de Prácticas 	2 horas
4	Comprendiendo el uso de estructuras repetitivas	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la 	<ul style="list-style-type: none"> • Apuntes de clase • Fuentes bibliográficas 	2 horas

		<p>práctica.</p> <ol style="list-style-type: none"> Realiza programas que resuelvan los problemas planteados por el docente. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> Computadora con Compilador de lenguaje C Manual de Prácticas 	
UNIDAD IV				
5	Funciones	<ol style="list-style-type: none"> Atiende las indicaciones del docente para para codificar los programas Codifica y prueba los ejemplos básicos que incluyen funciones, presentados por el docente. Realiza modificación a los ejemplos básicos de acuerdo a las indicaciones del docente. Entrega al docente, códigos modificados para su evaluación y retroalimentación. 	<ul style="list-style-type: none"> Computadora Compilador de C Documentación de biblioteca ANSI C 	2 horas
6	Funciones que retornan valores	<ol style="list-style-type: none"> Atiende las indicaciones del docente para para codificar los programas Codifica y prueba los ejemplos con funciones que retornan valores, presentados por el docente. Realiza modificación a los ejemplos básicos de acuerdo a las indicaciones del docente. Entrega al docente, códigos modificados para su evaluación y 	<ul style="list-style-type: none"> Computadora Compilador de C Documentación de biblioteca ANSI C 	2 horas

		retroalimentación.		
7	Variables: inicialización, variables estáticas, variables tipo registro y alcance de las variables.	<ol style="list-style-type: none"> 1. Atiende las indicaciones del docente para para codificar los programas 2. Codifica y prueba los ejemplos acerca de variables: inicialización, estáticas, tipo registro y alcance de las variables, presentados por el docente. 3. Realiza modificación a los ejemplos básicos de acuerdo a las indicaciones del docente. 4. Entrega al docente, códigos modificados para su evaluación y retroalimentación. 	<ul style="list-style-type: none"> • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	4 horas
UNIDAD V				
8	Aplicación de conceptos básicos de apuntadores	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real industrial o científico 2. Diseña e implementa uno o más soluciones utilizando conceptos básicos de apuntadores. 3. Redacta un reporte detallado que entregará al docente. 	<ul style="list-style-type: none"> • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas
	Utilizando apuntadores para eficientizar la memoria	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real industrial o científico. 2. Diseña e implementa uno o más soluciones utilizando arreglos de apuntadores. 3. Redacta un reporte detallado 	<ul style="list-style-type: none"> • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas

		que entregará al docente.		
9	Aplicación de conceptos avanzados de apuntadores	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real industrial o científico. 2. Diseña e implementa uno o más soluciones utilizando conceptos avanzados de apuntadores. 3. Redacta un reporte detallado que entregará al docente. 	<ul style="list-style-type: none"> • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas
UNIDAD VI				
10	Estructuras	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real del tipo industrial o científico. 2. Diseña, implementa y prueba una o más soluciones que hacen uso de agrupación de datos en estructuras. 3. Redacta un reporte detallado que entregará al docente. 	<ul style="list-style-type: none"> • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas
11	Memoria Dinámica	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real del tipo industrial o científico. 2. Diseña, implementa y prueba una o más soluciones que hacen uso de memoria dinámica. 3. Redacta un reporte detallado que entregará al docente. 	<ul style="list-style-type: none"> • Apuntes de clase y taller • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas
12	Uniones y Campos de bits	<ol style="list-style-type: none"> 1. Analiza uno o más problemas de aplicación real del tipo industrial o científico. 2. Diseña, implementa y prueba 	<ul style="list-style-type: none"> • Apuntes de clase y taller • Computadora • Compilador de C • Documentación de biblioteca 	2 horas

		<p>una o más soluciones que hacen uso de uniones y campos de bits.</p> <p>3. Redacta un reporte detallado que entregará al docente.</p>	ANSI C	
UNIDAD VII				
13	Uso de archivos secuenciales de texto	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la práctica. 2. Realiza programas que resuelvan los problemas planteados por el docente y que requieran de la lectura y escritura de archivos de texto. 3. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase y taller • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas
14	Uso de archivos de acceso aleatorio	<ol style="list-style-type: none"> 1. Atiende las orientaciones del profesor para elaborar la práctica. 2. Realiza programas que resuelvan los problemas planteados por el docente y que requieran de la lectura y escritura binarios de acceso aleatorio. 3. Entrega reporte de la práctica al docente para su retroalimentación. 	<ul style="list-style-type: none"> • Apuntes de clase • Computadora • Compilador de C • Documentación de biblioteca ANSI C 	2 horas

VII. MÉTODO DE TRABAJO

Encuadre: El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

Estrategia de enseñanza (docente):

- Técnica expositiva
- Retroalimentación
- Guía prácticas de taller y laboratorio
- Fomenta el trabajo colaborativo
- Plantea escenarios de aplicación del lenguaje C en el desarrollo de diversos problemas
- Realiza evaluaciones
- Guía en portafolio de evidencias

Estrategia de aprendizaje (alumno):

- Codificar programas de software
- Búsqueda de información
- Búsqueda de escenario de uso de las estructuras de datos
- Realiza prácticas de taller y laboratorio
- Trabaja de manera colaborativa
- Analiza la solución a problemas utilizando lenguaje C
- Diseña programa de software
- Redacta reportes de prácticas
- Realiza portafolio de evidencias

VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

Criterios de acreditación

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

Criterios de evaluación

- Tareas y prácticas de taller..... 20%
- Exámenes parciales..... 40%
- Portafolio de evidencias (prácticas)..... 40%
- Total.....100%**

IX. REFERENCIAS

Básicas

- Amini, K. (2019). *Extreme C*. Estados Unidos: Packt.
- Balagurusamy, E. (2019). *Programming In Ansi C*. McGraw Hill. India
- Seacord, R.C. (2020). *Effective C: An Introduction to Professional C Programming*. Estados Unidos: No Starch Press.
- Kernighan, B. & Ritchie, D. (1988). *The C programming language*. Englewood Cliffs, Estados Unidos: Prentice Hall. [clásica]

Complementarias

- Aguilar, L. & Martínez, I. (2014). *Programación en C, C++ JAVA y UML*. México: McGraw-Hill. [clásica]
- C/C++ Users Journal (2006). Estados Unidos: CMP Media, Inc. <https://dl.acm.org/journal/ccuj> .[clásica]
- Deitel, P. & Deitel, H. (2016). *C : how to program*. Estados Unidos: Pearson.
- Griffiths, D. & Griffiths, D. (2015). *Head First C*. Estados Unidos: O'Reilly Media, Inc.[clásica]

X. PERFIL DEL DOCENTE

El docente que imparta la asignatura deberá contar con una formación en ciencias de la computación o área afín, preferentemente con Maestría o Doctorado en área afín. Es deseable contar con experiencia docente mínima de dos años en la enseñanza de programación estructurada en nivel superior, así como haber aprobado cursos de formación docente.