



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

ÁREA: Ciencias de la Computación

ASIGNATURA: Fundamentos de Lenguajes de Programación

CÓDIGO: CCOS 255

CRÉDITOS: 6

FECHA: 14 de noviembre de 2022

Fundamentos de Lenguajes de Programación

PE: Licenciatura en Ciencias de la Computación



“El presente documento es Propiedad Intelectual de la Benemérita Universidad Autónoma de Puebla, conforme a lo previsto en el artículo 8 de su Ley y 137 del Estatuto Orgánico Universitario. La utilización del mismo, es para uso exclusivo de la Benemérita Universidad Autónoma de Puebla y los integrantes de la comunidad universitaria, en cumplimiento de los fines de docencia, investigación y extensión de la cultura. Queda estrictamente prohibida la reproducción total o parcial de su contenido o cualquier uso, distintos a los señalados en el párrafo anterior”.



Nivel Educativo:	Licenciatura.
Nombre del Plan de Estudios:	Licenciatura en Ciencias de la Computación
Modalidad Académica:	Presencial.
Nombre de la Asignatura:	Fundamentos de Lenguajes de Programación
Ubicación:	Nivel Formativo.
Correlación:	
Asignaturas Precedentes:	Lenguajes Formales y Autómatas
Asignaturas Consecuentes:	N/A

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)	5	0	90	6

Fundamentos de Lenguajes de Programación

PE: Licenciatura en Ciencias de la Computación



“El presente documento es Propiedad Intelectual de la Benemérita Universidad Autónoma de Puebla, conforme a lo previsto en el artículo 8 de su Ley y 137 del Estatuto Orgánico Universitario. La utilización del mismo, es para uso exclusivo de la Benemérita Universidad Autónoma de Puebla y los integrantes de la comunidad universitaria, en cumplimiento de los fines de docencia, investigación y extensión de la cultura. Queda estrictamente prohibida la reproducción total o parcial de su contenido o cualquier uso, distintos a los señalados en el párrafo anterior”.



Autores:	Claudia Zepeda Cortés, Alba Maribel Sánchez Gálvez, Meliza Contreras González, Mireya Tovar Vidal, César Bautista Ramos, José Raymundo Marcial Romero, Alfonso Garcés Báez, José de Jesús Lavalle Martínez.
Fecha de diseño:	Noviembre de 2009
Fecha de la última actualización:	11 de noviembre de 2022
Fecha de aprobación por parte de la academia de área, departamento u otro.	25 de noviembre de 2022
Revisores:	Claudia Zepeda Cortés, Alfonso Garcés Báez, Mireya Tovar Vidal, José de Jesús Lavalle Martínez
Sinopsis de la revisión y/o actualización:	Revisión y actualización de la bibliografía.

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

Disciplina profesional:	Licenciado o Ingeniero en electrónica, o Ingeniero en computación.
Nivel académico:	Maestría o superior.

Fundamentos de Lenguajes de Programación

PE: Licenciatura en Ciencias de la Computación



“El presente documento es Propiedad Intelectual de la Benemérita Universidad Autónoma de Puebla, conforme a lo previsto en el artículo 8 de su Ley y 137 del Estatuto Orgánico Universitario. La utilización del mismo, es para uso exclusivo de la Benemérita Universidad Autónoma de Puebla y los integrantes de la comunidad universitaria, en cumplimiento de los fines de docencia, investigación y extensión de la cultura. Queda estrictamente prohibida la reproducción total o parcial de su contenido o cualquier uso, distintos a los señalados en el párrafo anterior”.





Experiencia docente:	1 año.
Experiencia profesional:	3 años.

5. PROPÓSITO:

Revisar los conceptos sobre los fundamentos de lenguajes de programación para analizar, formalizar y resolver problemas reales que se plantean en las Ciencias de la Computación.

6. COMPETENCIAS PROFESIONALES:

- 1. Tener un conocimiento profundo de los principios fundamentales de las computadoras y del software, que le permita evaluar la complejidad de un problema de computación y recomendar las máquinas, lenguajes y paradigmas de programación más adecuados para diseñar e implementar una buena solución computacional.*
- 2. Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema computacional en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes o entornos inteligentes.*
- 3. Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes.*
- 4. Aplicar fundamentos matemáticos, principios algorítmicos y teorías de Ciencias de la Computación para el modelado y diseño de soluciones computacionales.*
- 5. Aplicar técnicas de investigación para la búsqueda, fundamentación y desarrollo de soluciones computacionales.*



Unidad de Aprendizaje	Contenido Temático	Referencias
1. Cálculo lambda tipificado	1.1 Sintaxis. 1.2 Mecanismo de abstracción lambda. 1.3 Algoritmo de sustitución para evaluar expresiones lambda. 1.4 Semántica operacional. (6 semanas)	Barendregt H., Dekkers W., Statman R. (2013). Lambda Calculus with Types (Perspectives in Logic). UK: Cambridge University Press. Nederpelt R., Geuvers H.(2014). Type Theory and Formal Proof: An Introduction. UK: Cambridge University Press. Gunter, C. A.(1992). Semantics of Programming languages: Structures and Techniques. UK: MIT Press. Hankin, C. (2004). An introduction to Lambda Calculi for Computer Science. UK: King's College London. Gordon M. (2013) The Denotational Description of Programming Languages: An Introduction. USA: Springer. MITOPENCOURSEWARE Massachusetts Institute of technology (Cursos en línea: goo/CI3GG6) Michaelson G. (). An Introduction to Funcional Programming Through Lambda Calculus. Dover Publications, INC. Harper R. (2016). Practical Foundations For



Unidad de Aprendizaje	Contenido Temático	Referencias
		Programming Languages. Cambridge University Press. Chacón, C., (2021). Computación y programación funcional. Edición Kindle, Ed. Marcombo

Unidad de Aprendizaje	Contenido Temático	Referencias
2. Un modelo para el cálculo lambda	2.1 Motivación acerca del papel que tiene la teoría de modelos al asignar significado a expresiones. 2.2 Modelo basado en conjuntos (Conjuntos denotados por expresiones lambda tipificadas). 2.3 Equivalencia entre la semántica operacional y el modelo basado en conjuntos (Relaciona las reglas operacionales con sus correspondientes mapeos entre conjuntos). (6 semanas)	Barendregt H., Dekkers W., Statman R. (2013). Lambda Calculus with Types (Perspectives in Logic). UK: Cambridge University Press. Nederpelt R., Geuvers H.(2014). Type Theory and Formal Proof: An Introduction. UK: Cambridge University Press. Gunter, C. A.(1992). Semantics of Programming languages: Structures and Techniques. UK: MIT Press. Hankin, C. (2004). An introduction to Lambda Calculi





Unidad de Aprendizaje	Contenido Temático	Referencias
		for Computer Science. UK: King's College London. Gordon M. (2013) The Denotational Description of Programming Languages: An Introduction. USA: Springer. MITOPENCOURSEWARE Massachusetts Institute of technology (Cursos en línea: goo/CI3GG6) Michaelson G. (). An Introduction to Funcional Programming Through Lambda Calculus. Dover Publications, INC. Harper R. (2016). Practical Foundations For Programming Languages. Cambridge University Press. Chacón, C., (2021). Computación y programación funcional. Edición Kindle, Ed. Marcombo





Unidad de Aprendizaje	Contenido Temático	Referencias
3. Un lenguaje para programar	3.1 Sintaxis 3.2 Semánticas axiomática y operacional 3.3 Estrategias de reducción (6 semanas)	Mitchell, J. C. Foundations for Programming Languages. The MIT Press. 1996. Barendregt H., Dekkers W., Statman R. (2013). Lambda Calculus with Types (Perspectives in Logic). UK: Cambridge University Press. Nederpelt R., Geuvers H.(2014). Type Theory and Formal Proof: An Introduction. UK: Cambridge University Press. Gunter, C. A.(1992). Semantics of Programming languages: Structures and Techniques. UK: MIT Press. Hankin, C. (2004). An introduction to Lambda Calculi for Computer Science. UK: King's College London. Gordon M. (2013) The Denotational Description of Programming Languages: An Introduction. USA: Springer. MITOPENCOURSEWARE Massachusetts Institute of technology (Cursos en línea: goo/CI3GG6) Michaelson G. (). An Introduction to Funcional Programming Through Lambda Calculus. Dover Publications, INC.





Unidad de Aprendizaje	Contenido Temático	Referencias
		Harper R. (2016). Practical Foundations For Programming Languages. Cambridge University Press. Chacón, C., (2021). Computación y programación funcional. Edición Kindle, Ed. Marcombo

8. ESTRATEGIAS, TÉCNICAS Y RECURSOS DIDÁCTICOS

Estrategias y técnicas didácticas	Recursos didácticos
Lluvia o tormenta de ideas Técnica de debate Método de casos Estado del arte Mapas mentales Grupos de discusión Solución de problemas Aprendizaje basado en problemas Aprendizaje basado en proyectos	Impresos (textos): libros, fotocopias, documentos, artículos. Materiales audiovisuales: Videos o películas. Imágenes fijas proyectables (fotos)-diapositivas, fotografías. Programas informáticos (CD u on-line) educativos: videojuegos, presentaciones multimedia, enciclopedias, animaciones. Páginas Web, correo electrónico, chats, foros y cursos on-line

Fundamentos de Lenguajes de Programación

PE: Licenciatura en Ciencias de la Computación



“El presente documento es Propiedad Intelectual de la Benemérita Universidad Autónoma de Puebla, conforme a lo previsto en el artículo 8 de su Ley y 137 del Estatuto Orgánico Universitario. La utilización del mismo, es para uso exclusivo de la Benemérita Universidad Autónoma de Puebla y los integrantes de la comunidad universitaria, en cumplimiento de los fines de docencia, investigación y extensión de la cultura. Queda estrictamente prohibida la reproducción total o parcial de su contenido o cualquier uso, distintos a los señalados en el párrafo anterior”.





Eje (s) transversales	Contribución con la asignatura
Formación Humana y Social	Facilitando la comprensión de los diferentes modelos de computación con sus alcances y limitaciones haciendo uso del pensamiento crítico, el análisis y la reflexión, además del respeto a las ideas de otros.
Desarrollo de Habilidades en el uso de las Tecnologías de la Información y la Comunicación	Promoviendo el uso de cursos en línea tales como los de MIT, Harvard., revisando tutoriales y software que facilite el aprendizaje del material visto en clase.
Desarrollo de Habilidades del Pensamiento Complejo	Integrando los conocimientos previos, los generalizándolos y aplicándolos a casos concretos.
Lengua Extranjera	Facilitando la comprensión de la bibliografía, pues mucha de ésta se encuentra en Inglés.
Innovación y Talento Universitario	Fomentando la creatividad a la solución de problemas de forma novedosa y elegante.
Educación para la Investigación	Buscando, proponiendo, corroborando y sistematizando constantemente en el curso.





Criterios	Porcentaje
▪ Exámenes	40%
▪ Tareas	10%
▪ Programas	20%
▪ Proyecto final	30%
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
Asistir como mínimo al 70% de las sesiones para tener derecho al examen extraordinario
Cumplir con las actividades académicas y cargas de estudio asignadas que señale el PE

Notas:

- 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.
- La planeación didáctica deberá ser entregada a la coordinación de la licenciatura en los tiempos y formas acordados por la Unidad Académica.