UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO
LICENCIATURA EN INGENIERÍA EN INTELIGENCIA ARTIFICIAL

PROGRAMA DE ESTUDIOS

PROGRAMACIÓN I

Elaboró:	Dra. Silvia Edith Albarrán Trujillo	
	Ing. Francisco Chávez Castañeda	Facultad de Ingeniería
	Mtra. Sara Vera Noguez	
Asesoría técnica:	Mtra. Alicia Celen Macedo Delgado	Dirección de Estudios Profesionales

	H. Consejo Académico	H. Consejo de Gobierno
Fecha de aprobación:	25 de abril de 2024	25 de abril de 2024

Facultad de Ingeniería

I. Datos de identificación.

Espacio académico donde se imparte	**Facultad de Ingeniería**						
Estudios profesionales	**Licenciatura en Ingeniería en Inteligencia Artificial**			Año de aprobación	**2024**		
Unidad de aprendizaje	**Programación I**						
Carga académica	**4** Horas teóricas	**0** Horas prácticas	**4** Total de horas	**8** Créditos			
Carácter	**Obligatoria**	Tipo	**Curso**	Periodo escolar	**Primero**		
Área curricular	**Ciencias de la Ingeniería**			Núcleo de formación	**Básico**		
Seriación	**Ninguna** UA Antecedente			**Programación II** UA Consecuente			

Formación común

No presenta **X**

II. Presentación del programa de estudios.

La programación es un habilitador fundamental para los ingenieros en inteligencia artificial, ya que les enseña a pensar de manera lógica; les permite implementar algoritmos, desarrollar modelos, optimizar rendimiento y resolver problemas de manera estructurada en este campo en constante evolución; así como, diseñar y traducir algoritmos teóricos en código funcional que puede ser ejecutado por computadoras.

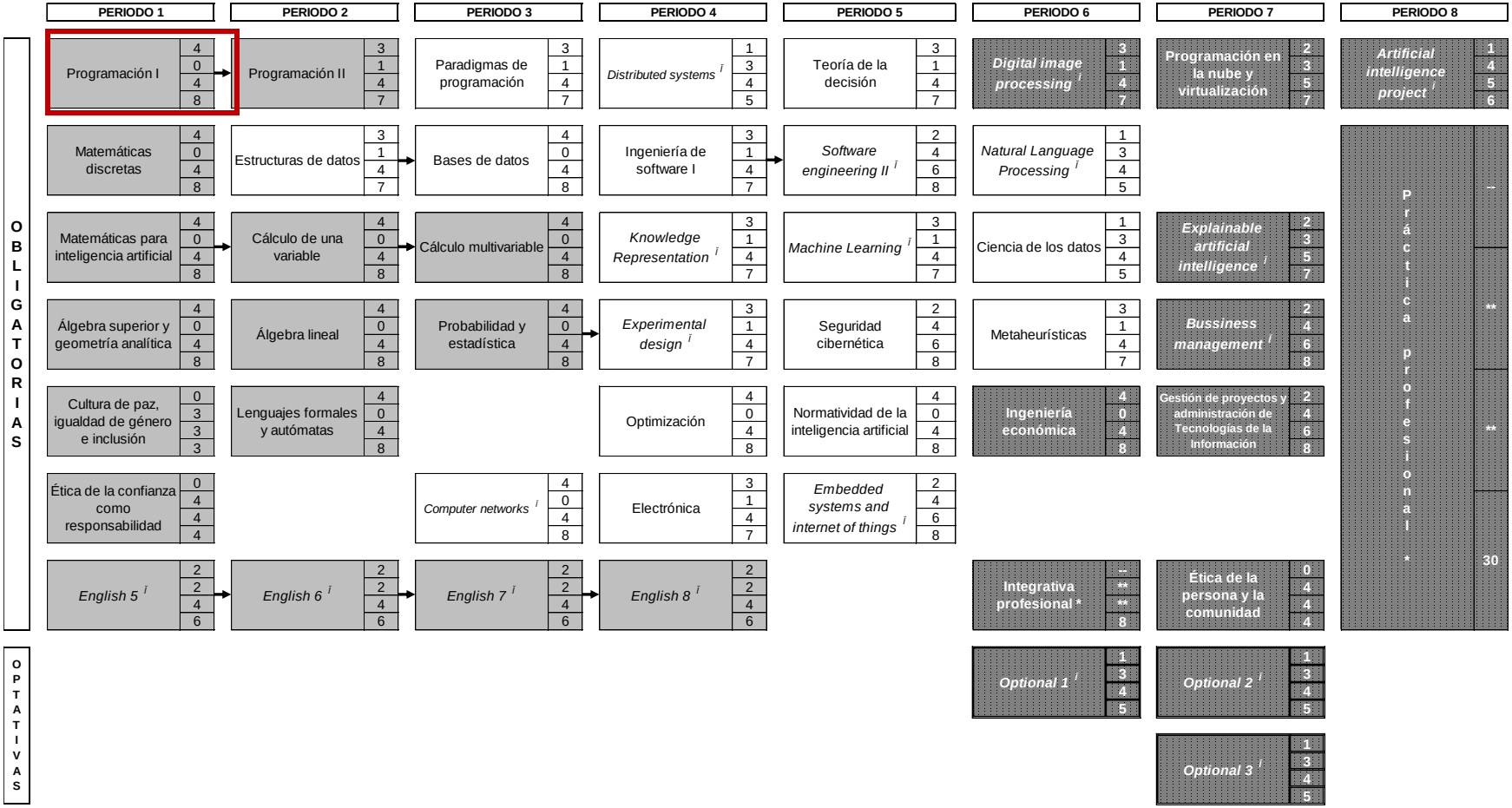
En la inteligencia artificial se requiere programar algoritmos complejos de aprendizaje automático, redes neuronales, lógica difusa, algoritmos genéticos, entre otros. Así, la construcción de modelos implica escribir código para entrenarlos con datos, ajustar parámetros y validar su desempeño. Esto requiere habilidades de programación para implementar eficientemente estos procesos.

Además, la integración de sistemas de IA en aplicaciones y entornos del mundo real requiere habilidades de programación para conectar y comunicar diferentes componentes de software, bases de datos y servicios en la nube.

Por tanto, este primer curso (Programación I) permite adquirir conocimientos que sirven de base para Programación II, Estructuras de datos, Bases de datos e Ingeniería de software; y busca llevar al estudiante desde un panorama del desarrollo de software hasta su implementación, brindándole elementos básicos de los programas, estrategias para diseñar algoritmos y usar herramientas para elaborar programas

III. Ubicación de la unidad de aprendizaje en el mapa curricular

LICENCIATURA EN INGENIERÍA EN INTELIGENCIA ARTIFICIAL, 2024



HT	18	HT	20	HT	21	HT	19
HP	9	HP	4	HP	3	HP	9
TH	27	TH	24	TH	24	TH	28
CR	45	CR	44	CR	45	CR	47

HT	16	HT	13	HT	10	HT	1
HP	14	HP	11+**	HP	24	HP	4+**
TH	30	TH	24+**	TH	34	TH	5+**
CR	46	CR	45	CR	44	CR	36

DISTRIBUCIÓN DE LAS UNIDADES DE APRENDIZAJE OPTATIVAS

PERIODO 1	PERIODO 2	PERIODO 3	PERIODO 4	PERIODO 5	PERIODO 6	PERIODO 7	PERIODO 8
					Wearable computing [I] 1 3 4 5	*Reinforcement learning* [I] 1 3 4 5	
					Pattern recognition [I] 1 3 4 5	*Generative artificial intelligence* [I] 1 3 4 5	
					Artificial neural networks and deep learning [I] 1 3 4 5	*Big Data* [I] 1 3 4 5	
					Evolutionary computation [I] 1 3 4 5	*Computer vision* [I] 1 3 4 5	
					3D Visualization [I] 1 3 4 5	*Automatic text processing* [I] 1 3 4 5	
					Robot programming [I] 1 3 4 5	*Domotic and smart cities* [I] 1 3 4 5	
						Human-Computer Interaction [I] 1 3 4 5	

SIMBOLOGÍA

Unidad de aprendizaje (UA)	HT: Horas Teóricas
	HP: Horas Prácticas
	TH: Total de Horas
	CR: Créditos

➔ 9 líneas de seriación.

Créditos mínimos 22 y máximos 47 por periodo escolar.

*Actividad académica.

**Las horas de la actividad académica.

[I] UA optativa que debe impartirse, cursarse y acreditarse en la lengua inglesa.

PARÁMETROS DEL PLAN DE ESTUDIOS

Núcleo básico: cursar y acreditar 16 UUAA obligatorias	47
	16
	63
	110
Núcleo sustantivo: cursar y acreditar 19 UUAA obligatorias	52
	30
	82
	134
Núcleo integral: cursar y acreditar 8 UUAA + 2* obligatorias	16
	23+**
	39+**
	93
Núcleo integral: cursar y acreditar 3 UUAA optativas	3
	9
	12
	15

Total del núcleo básico: acreditar 16 UUAA para cubrir 110 créditos

Total del núcleo sustantivo acreditar 19 UUAA para cubrir 134 créditos

Total del núcleo integral acreditar 11 UUAA + 2 * para cubrir 108 créditos

TOTAL DEL PLAN DE ESTUDIOS	
UUAA obligatorias	43 + 2 Actividad académica
UUAA optativas	3
UUAA a acreditar	46 + 2 Actividad académica
Créditos	352

IV. Objetivos de la formación profesional.

Objetivos del programa educativo:

La licenciatura en Ingeniería en Inteligencia Artificial tiene como propósito formar profesionales competentes y éticos con una formación integral y sólida para:

- Examinar las aplicaciones de las principales áreas de la inteligencia artificial como el aprendizaje automático, el procesamiento de lenguaje natural, la visión por computadora, el razonamiento automatizado y el internet de las cosas, mediante el estudio de teorías y modelos matemáticos y computacionales para comprender y discernir su adecuación en soluciones basadas en IA de problemas complejos.
- Analizar problemas que la IA puede resolver con técnicas de aprendizaje automático, procesamiento de lenguaje natural, visión por computadora y razonamiento automatizado, así como problemáticas futuras para anticiparse a retos tecnológicos y científicos de la disciplina y desarrollar soluciones innovadoras en áreas de salud, finanzas, negocios, educación, etc.
- Crear soluciones basadas en inteligencia artificial mediante el diseño utilizando métricas diseño, implementación y evaluación de sistemas técnicos, éticos, sociales, legales, ambientales y económicos, para abordar y resolver problemas complejos promoviendo la sostenibilidad y el bienestar social.
- Evaluar sistemas de inteligencia artificial utilizando métricas de rendimiento, técnicas de validación, análisis de errores y consideraciones éticas para generar soluciones eficientes, transparentes, viables y libres de sesgos.
- Fomentar la creatividad y el pensamiento crítico analizando los resultados e implicaciones del uso de la inteligencia artificial para el diseño de soluciones tecnológicas innovadoras, eficientes, éticas y responsables con el entorno y la sociedad

Objetivos del núcleo de formación:

Promoverá en el alumno el aprendizaje de las bases contextuales, teóricas y filosóficas de sus estudios, la adquisición de una cultura universitaria en las ciencias y las humanidades, y el desarrollo de las capacidades intelectuales indispensables para la preparación y ejercicio profesional, o para diversas situaciones de la vida personal y social.

Objetivos del área curricular o disciplinaria:

Analizar los fundamentos de la programación, ingeniería de software, redes, electrónica, bases de datos y seguridad cibernética mediante la aplicación de técnicas, herramientas, metodologías en el diseño, implementación y gestión de sistemas hardware y software para desarrollar soluciones innovadoras, escalables y eficientes a problemas complejos con requerimientos de datos y automatización de procesos.

V. Objetivos de la unidad de aprendizaje.

Crear algoritmos utilizando razonamiento formal a través del modelado, elementos de un programa, técnicas de elaboración e implementación de algoritmos y pruebas de escritorio para la solución de problemas.

VI. Contenidos de la unidad de aprendizaje, y su organización.

Unidad temática 1. Modelado
Objetivo: Identificar los pasos y técnicas en el desarrollo de algoritmos mediante el estudio del modelo de cascada para la solución de problemas implementables por una computadora.
Temas: 1.1 Modelo de cascada 1.2 Identificación de requerimientos 1.3 Identificación de escenarios 1.4 Estrategias de diseño 1.4.1 Divide y vencerás 1.4.2 Diseño descendente (Top-Down)

Unidad temática 2. Elementos de un programa
Objetivo: Distinguir los elementos de un programa en el desarrollo de algoritmos mediante el uso de los conceptos del modelo de Von Newman para la solución de problemas implementables por una computadora.
Temas: 2.1 Modelo de Von Newman 2.2 Tipos de variables (primitivas y de referencia) 2.3 Tipos de datos 2.4 Operadores 2.5 Estructuras de control 2.6 Funciones

2.7 Uso de objetos 2.8 Operaciones con objetos

Unidad temática 3. Elaboración de algoritmos
Objetivo: Integrar técnicas de modelado y el diseño descendente, mediante el uso de diagramas UML para el diseño de sistemas simples.
Temas: 3.1 UML 3.2 Diagrama de secuencias 3.3 Diagrama de actividades 3.4 Aplicación del diseño descendente 3.5 Plan de pruebas 3.6 Pruebas de escritorio

Unidad temática 4. Implementación de algoritmos
Objetivo: Formular soluciones utilizando un lenguaje de programación y herramientas de desarrollo integradas (IDE) para la implementación de sistemas simples.
Temas: 4.1 Lenguaje de programación 4.1.1 Estructura del programa en el lenguaje 4.1.2 Sintaxis 4.2 Uso del entorno de programación 4.3 Manejo de excepciones 4.4 Depuración de programas

VII. Acervo bibliográfico.

Básico:

Hunt, J. (2023). *A Beginners Guide to Python 3 Programming*. Springer International Publishing.

Mastrodomenico, R. (2022). *The Python Book*. Wiley and Sons.

Romano, F., Kruger, H. (2021). Learn Python Programming: An In-Depth Introduction to the Fundamentals of Python (3rd Edition). Packt Publishing.

Severance, C. R. (2016). Python for Everybody: Exploring Data Using Python 3. Charles Severance.Youens.

Clark, K. (2020). Tiny Python Projects: 21 Small Fun Projects for Python Beginners Designed to Build Programming Skill, Teach New Algorithms and Techniques, and Introduce Software Testing. Manning.

Complementario:

Asif, M. (2021). *Python for Geeks: Build Production-ready Applications Using Advanced Python Concepts and Industry Best Practices* Packt Publishing.

Booch, G., Rumbaugh, J., Jacobson, I. (2000). *The Complete UML Training Course*. Prentice Hall PTR.

Debrauwer, L.,Van der Heyde, F. (2016). *UML 2.5: iniciación, ejemplos y ejercicios corregidos.* ENI.

Downey, A. B. (2012). *Think Python*. O'Reilly Media.

Pirnat, M. (2015). *How to Make Mistakes in Python*. O'Reilly Media.

Vanderplas, J. T. (2016). *A Whirlwind Tour of Python*. O'Reilly Media.

Zelle, J. M. (2004). *Python Programming: An Introduction to Computer Science.* Franklin, Beedle.

Zhu, H. (2005). *Software Design Methodology: From Principles to Architectural Styles*. Países Bajos: Elsevier Science.