

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO
LICENCIATURA EN ACTUARÍA



GUÍA DE EVALUACIÓN DEL APRENDIZAJE

PROGRAMACIÓN

Elaboró:	M. en M.A. Verónica Ángeles Morales	Facultad de Economía
	I. S. C. Ivette del Rosario Cruz Solís	Facultad de Economía
	M. en C. Mario Alberto Hernández Becerril	Facultad de Economía

Fecha de aprobación:	H. Consejo Académico	H. Consejo de Gobierno
	30 de agosto de 2021	30 de agosto de 2021

Facultad de Economía



Índice

	Pág.
I. Datos de identificación.	3
II. Presentación de la Guía.	4
III. Ubicación de la unidad de aprendizaje en el mapa curricular.	5
IV. Objetivos de la unidad de aprendizaje.	8
V. Diseño de la evaluación: Factores, Criterios e Indicadores	8
VI. Diseño de los instrumentos de observación	13
a) Mediciones que derivan en puntajes	13
b) Estimaciones no cuantificables	15
VII. Administración de los instrumentos y registro de evidencias	16
VIII. Evaluación del aprendizaje	19
a) Interpretación de apreciaciones y/o datos	19
b) Juicios y conclusiones valorativas	19
c) Asignación, entrega y revisión de resultados	20



I. Datos de identificación.

Espacio académico donde se imparte **Facultad de Economía
Centro Universitario UAEM Valle de México
Unidad Académica Profesional Cuautitlán Izcalli
Unidad Académica Profesional Huehuetoca**

Estudios profesionales **Licenciatura en Actuaría, 2021**

Unidad de aprendizaje **Programación**

Carga académica

1	5	6	7
Horas teóricas	Horas prácticas	Total de horas	Créditos

Carácter **Obligatoria** Tipo **Taller** Periodo escolar **Primero**

Área curricular **Ciencias Exactas** Núcleo de formación **Básico**

Seriación **Ninguna** **Bases de datos**
UA Antecedente UA Consecuente

Formación común

Licenciatura	Economía, 2021	X
	Actuaría, 2021	X
	Relaciones Económicas Internacionales, 2021	



II. Presentación de la Guía.

El presente documento se elabora como complemento normativo del Programa de Estudios de la Unidad de Aprendizaje de Programación que corresponde al primer semestre de la licenciatura en Actuaría. La guía expone los criterios, instrumentos y procedimientos establecidos para la evaluación del Programa de Estudios, que será a través de dos exámenes parciales con el valor del 30% en cada examen. Además se integran actividades que complementan la aplicabilidad de los temas de cada unidad, éstas serán evaluadas a través de listas de cotejo y rúbricas (con diferentes ponderaciones) con un valor total del 70% de la calificación por cada parcial y un producto final (100%). Los exámenes e instrumentos tienen como objetivo evaluar los conocimientos relativos a: soluciones de problemas computacionales, esquemas de algoritmos y pseudocódigo, el contexto del lenguaje C y C++, que incluye estructura de control, cíclicas, funciones y procedimiento. También las estructuras de los esquemas de apuntadores, y el manejo de clases, objetos, registros y archivos; que den solución óptima a problemas económicos y financieros.

Adicionalmente, el proceso de evaluación de *Programación* deberá considerar al menos tres indicadores que evalúen el desempeño del alumno en algunos temas representativos de la primera parte de la unidad de aprendizaje; mismos que se evaluarán a través de una lista de cotejo hasta antes de la primera evaluación parcial. Dicha evaluación debe ser continua incorporando prácticas y ejercicios que den soporte al seguimiento de las competencias que se deben desarrollar en esta unidad de aprendizaje; en virtud de que el aprendizaje de Programación exige la práctica recurrente de sus diversos procedimientos o métodos, resulta de gran utilidad, para dicho fin, la resolución constante de ejercicios aplicados de los diversos temas estudiados en dicha unidad de aprendizaje, dando como resultado un mejor desarrollo de los productos a evaluar mediante rúbricas. Esto último resultará de gran utilidad en la preparación misma del estudiante para la evaluación final.

Esta guía pretende ser de apoyo para los docentes y alumnos, en la cual se transparentan las actividades, instrumentos y productos que servirán como evidencias bajo los cuales se sustentará la evaluación del curso durante todo el semestre, para con ello, realizar de manera objetiva y justa dichas actividades, valorando el aprendizaje obtenido.

La guía puntualiza los factores, criterios e indicadores por cada unidad, así como su esquema con respecto al tipo de evaluación y el instrumento aplicado en cada indicador.



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

MAPA CURRICULAR DE LA LICENCIATURA EN ACTUARÍA, 2021

	PERIODO 1	PERIODO 2	PERIODO 3	PERIODO 4	PERIODO 5	PERIODO 6	PERIODO 7	PERIODO 8	PERIODO 9	PERIODO 10	
O	Cálculo diferencial de una variable	Cálculo integral de una variable	Cálculo diferencial de varias variables	Cálculo integral de varias variables	Análisis matemático	Análisis numérico	Análisis de regresión	Serías de tiempo	Modelos y simulación	Práctica profesional	
	Álgebra superior	Álgebra lineal	Microeconomía	Ecuaciones diferenciales	Inferencia estadística	Estadística matemática	Teoría del riesgo	Estadística bayesiana	Modelos actuariales		
											Programación
	Contabilidad financiera	Mercados financieros	Teoría de la probabilidad	Probabilidad avanzada	Contabilidad de seguros y fianzas	Procesos estocásticos	Investigación de operaciones determinística	Administración de riesgos	Demografía matemática		
											Geometría analítica
	Matemáticas financieras avanzadas	Inglés 5	Inglés 6	Inglés 7	Inglés 8	Optativa 1, línea de acentuación	Optativa 2, línea de acentuación	Optativa 3, línea de acentuación	Optativa 4, línea de acentuación		
HT 18											HT 12
B	2	2	2	2	3	3	3	2	1		---
L	4	4	4	4	2	2	2	2	5		**
I	6	6	6	6	6	6	6	6	6		**
G	8	8	8	8	8	8	8	8	7	**	
A	1	1	3	3	3	3	2	1	2	**	
T	5	5	1	2	2	2	4	4	2	**	
O	6	6	4	4	4	4	6	6	4	**	
R	7	7	7	8	8	8	8	6	6	**	
I	3	3	4	4	1	3	2	1	1	30	
A	1	2	2	2	3	2	4	4	3	1	
S	7	8	10	10	5	8	8	6	5	2	
O	4	0	3	4	3	2	2	0	1	1	
P	2	4	2	2	1	2	4	6	3	2	
T	6	4	5	6	4	4	6	6	4	2	
T	10	4	8	10	7	6	8	6	5	3	
T	4	2	2	2	2	2	2	0	0	4	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6	6	4	3	
T	10	6	6	6	7	6	8	6	4	4	
T	4	2	2	2	2	2	2	0	0	1	
T	2	2	2	2	3	2	4	6	4	2	
T	6	4	4	4	4	4	6				



DISTRIBUCIÓN DE LAS UNIDADES DE APRENDIZAJE OPTATIVAS

		PERIODO 1	PERIODO 2	PERIODO 3	PERIODO 4	PERIODO 5	PERIODO 6	PERIODO 7	PERIODO 8	PERIODO 9	PERIODO 10	
Línea de acentuación	Modelos matemáticos y probabilísticos						Teoría de Juegos 3 1 4 7	Diseño de experimentos 2 3 5 7	Investigación de operaciones heurística 2 3 5 7	Teoría de la medida 2 3 5 7		
	Análisis de datos						Bases de datos complejas 2 3 5 7	Estadística para el análisis de datos 2 3 5 7	Simulación de sistemas dinámicos 2 3 5 7	Stochastic Modeling ¹ 2 3 5 7		
	Ciencias actuariales						Auditoría actuarial 3 1 4 7	Reaseguro 3 1 4 7	Teoría del riesgo avanzada 3 1 4 7	Investigación de mercados 2 3 5 7	Administración de riesgos avanzada 3 1 4 7	
	Economía						Microeconomía avanzada 1 5 6 7	Microeconomía 3 1 4 7	Macroeconomía avanzada 2 3 5 7	Demographic models ¹ 1 5 6 7	Macroeconometrics ¹ 3 1 4 7	
	Finanzas cuantitativas						Finanzas internacionales 3 1 4 7	Ingeniería financiera 1 5 6 7	Evaluación social de proyectos de inversión 2 3 5 7	Economía del medio ambiente y sustentable 3 1 4 7	Finanzas públicas 3 1 4 7	Corporate finance ¹ 3 1 4 7



Proyecto curricular de la Licenciatura en Actuaría
Reestructuración, 2021

Dirección de Estudios Profesionales • Departamento de Desarrollo Curricular



SIMBOLOGÍA

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

- 12 líneas de seriación
- 22 créditos mínimos y 56 máximos por periodo escolar
- * Actividad académica.
- ** Las horas de la actividad académica
- ° La carga horaria de las UUAA optativas que curse y acredite el alumno.
- ¹ UA optativa que debe impartirse, cursarse y acreditarse en el idioma inglés.

PARÁMETROS DEL PLAN DE ESTUDIOS

Núcleo básico:	48
cursar y acreditar	53
19 UUAA	101
obligatorias	149

Núcleo sustantivo: cursar y acreditar 20 UUAA obligatorias	57
	47
	104
	161

Núcleo integral: cursar y acreditar 13 UUAA + 1* obligatorias	15
	50+**
	65+**
	110

Núcleo integral: elegir línea de acentuación para cursar y acreditar 4 UUAA optativas	°
	°
	°
	28

Total del núcleo básico: acreditar 19 UUAA para cubrir 149 créditos

Total del núcleo sustantivo: acreditar 20 UUAA para cubrir 161 créditos

Total del núcleo integral: acreditar 17 UUAA + 1* para cubrir 138 créditos

TOTAL DEL PLAN DE ESTUDIOS

UUAA Obligatorias	52 + 1 Actividad académica
UUAA Optativas	4
UUAA a Acreditar	56 + 1 Actividad académica
Créditos	448

DIRECCIÓN DE ESTUDIOS
PROFESIONALES



Departamento de Desarrollo Curricular

Guía de Evaluación del Aprendizaje
Aprobada por los HH. Consejos
Académico y de Gobierno



IV. Objetivos de la unidad de aprendizaje.

Emplear las herramientas, estructuras y funciones del lenguaje C/C++ y los elementos de la programación estructurada para construir códigos en un lenguaje orientado a objetos que den solución óptima a problemas económicos y financieros.

V. Diseño de la evaluación: Factores, Criterios e Indicadores.

Unidad 1. Elementos de programación y programación estructurada		
Factores	Criterios	Indicadores
<p>Simular problemas en la computadora a través de las estructuras de control selectivas y repetitivas, módulos y subrutinas, la elaboración de algoritmos y diagramas de flujo para la construcción de un programa</p> <p>Factor Metodológico</p>	<p>Resolución de problemas computacionales</p> <p>Diseño de algoritmo para solucionar problemas informáticos.</p> <p>Elaboración de pseudocódigo.</p> <p>Aplicación de estructura de control.</p>	<p>Diseña una solución computacional a problemas planteados a través de estructuras de control selectivas de forma ordenada y sistemática.</p> <p>Elabora un esquema algorítmico a través de estructuras de control repetitivas para simplificar y solucionar problemas bajo un esquema computacional de manera puntual y estructurado.</p> <p>Verifica a través de pruebas de escritorio las propuestas algorítmicas en la solución de problemas de manera precisa e individual.</p>



Unidad 2. Lenguaje C++		
Factores	Criterios	Indicadores
<p>Emplear pseudocódigos en el lenguaje de programación C++ a través de la formulación de programas computacionales, manejo de librerías de C++, para la resolución de ejercicios que den solución a un problema de la disciplina.</p> <p>Factor Metodológico</p>	<p>Creación de códigos de programación con lenguaje de programación C/C++</p> <p>Aplicación de tipos de datos en los códigos de programación</p> <p>Uso de operadores aritméticos en soluciones programables.</p>	<p>Soluciona problemas computacionales a través del lenguaje de programación C y C++, incorporando elementos significativos del lenguaje bajo un esquema estructurado y puntual.</p> <p>Desarrolla soluciones computacionales a través de programas en lenguaje C y C++, empleando la estructura completa y funcional del lenguaje.</p>

Unidad 3. Estructuras de control, funciones y procedimientos		
Factores	Criterios	Indicadores
<p>Resolver problemas computacionales a través de las funciones y herramientas de la programación estructurada en lenguaje C++ para la elaboración de programas optimizados con funciones avanzadas.</p> <p>Factor metodológico</p>	<p>Uso de estructuras de control y cíclicas</p> <p>Creación de funciones eficientes</p> <p>Programación para solucionar problemas</p>	<p>Diseña funciones específicas en lenguaje C++ para resolver problemas computacionales aplicando la optimización de código y la aplicación de toma de decisiones.</p> <p>Desarrolla funciones avanzadas para incorporar a los programas computacionales de forma sistemática, definidas y puntual en C++.</p>



Unidad 4. Estructuras de datos		
Factores	Criterios	Indicadores
<p>Analizar un método potente de almacenamiento de datos aprendiendo cómo agrupar datos similares en arrays o arreglos “listas y tablas” numéricas; así como la gestión dinámica de la memoria y los operadores asociados a esta tarea; las propiedades de herencia y polimorfismo; las estructuras de datos conocidas como pilas y colas; por medio del conocimiento de su estructura, diseño y manipulación de los algoritmos básicos para profundizar en la programación orientada a objetos.</p> <p>Factor Metodológico</p>	<p>Aplicación de arreglos unidimensionales, bidimensionales y multidimensionales en C++</p> <p>Uso de polimorfismo con funciones y métodos en C++</p> <p>Creación de pilas y colas para gestionar información en C++</p>	<p>Incorpora como solución estructuras de datos estáticas como los arreglos en problemas de gestión de datos en forma crítica y creativa en C++.</p> <p>Identifica correctamente algoritmos computacionales en procesos de ordenamiento o búsquedas de datos reduciendo costos de programación en la solución de problemas en C++.</p> <p>Aplica operaciones de pilas y colas a problemas computacionales, considerando las estructuras de datos dinámicas como éstas en soluciones basados en el paradigma de la programación orientada a objetos en C++.</p>

Unidad 5. Punteros (apuntadores)		
Factores	Criterios	Indicadores
<p>Analizar la eficiencia de los punteros (apuntadores) en el lenguaje C++ a través del conocimiento del concepto de puntero, los punteros null y void, punteros de cadenas, punteros como</p>	<p>Uso los punteros en funciones en C++</p> <p>Aplicación de punteros para solucionar problemas computacionales en C++</p>	<p>Establece pautas concretas en la definición y utilización de punteros en las soluciones de problemas computacionales, estableciendo criterios y funcionalidades de esta en los programas</p>



<p>argumentos de funciones, punteros a funciones, punteros a estructuras y la aritmética de punteros; para la construcción de programas eficientes y potentes.</p> <p>Factor Metodológico</p>		<p>desarrollados en C++.</p> <p>Crea programas aplicando punteros basados en la programación orientada a objetos de forma puntual y eficiente en C++.</p>
--	--	---

Unidad 6. Clases y objetos		
Factores	Criterios	Indicadores
<p>Estudiar conceptos fundamentales de la programación orientada a objetos: clases, objetos, constructores, destructores y clases compuestas, para complementar el uso de estructuras de datos complejas.</p> <p>Factor Teórico</p>	<p>Construcción de clases</p> <p>Aplicación de constructores, destructores y métodos en las clases</p> <p>Uso de clase exception para evitar errores de programación</p>	<p>Utiliza las clases como parte de las soluciones a problemas computacionales especificando los atributos y métodos de cada una de ellas y aplicando los objetos correctamente.</p> <p>Aplica la herencia como parte fundamental en las clases compuestas dado que se incorpora esquemas del paradigma de la programación orientada a objetos de forma sistemática y estructural.</p> <p>Incorpora en los códigos de programación las clases de exception para identificar y solucionar errores de programación de forma puntual y simplificada.</p>



Unidad 7. Registros y archivos

Factores	Criterios	Indicadores
<p>Analizar las estructuras de datos básicas de almacenamiento, en dispositivos externos: registros y archivos a través de su conceptualización y elementos básicos para la manipulación de grandes cantidades de información dando solución a problemas de complejos.</p> <p>Factor Teórico</p>	<p>Creación de procesos para la gestión de archivos</p> <p>Aplicación de operaciones de lectura y escritura en los archivos</p> <p>Uso correcto de los archivos secuenciales y de acceso aleatorio</p>	<p>Establece criterios de selección en el uso de archivos en la solución computacional, identificando la información procesada y su funcionalidad, de forma sistemática y formal.</p> <p>Propone esquemas de programación para la gestión de datos a través de archivos mediante la organización, tratamiento y mantenimiento de la información en ellos.</p>



VI. Diseño de los instrumentos de observación

a) Mediciones que derivan en puntajes

Indicador	Evaluación	Instrumento
Diseña una solución computacional a problemas planteados a través de estructuras de control selectivas de forma ordenada y sistemática.	Formativa	Lista de cotejo
Elabora un esquema algorítmico a través de estructuras de control repetitivas para simplificar y solucionar problemas bajo un esquema computacional de manera puntual y estructurado.	Formativa	Lista de cotejo
Verifica a través de pruebas de escritorio las propuestas algorítmicas en la solución de problemas de manera precisa e individual.	Formativa	Lista de cotejo
Soluciona problemas computacionales a través del lenguaje de programación C y C++, incorporando elementos significativos del lenguaje bajo un esquema estructurado y puntual.	Formativa	Rúbrica
Desarrolla soluciones computacionales a través de programas en lenguaje C y C++, empleando la estructura completa y funcional del lenguaje.	Formativa	Rúbrica
Diseña funciones específicas en lenguaje C++ para resolver problemas computacionales aplicando la optimización de código y la aplicación de toma de decisiones.	Formativa	Rúbrica
Desarrolla funciones avanzadas para incorporar a los programas computacionales de forma sistemática, definidas y puntual en C++	Formativa	Rúbrica



Incorpora como solución estructuras de datos estáticas como los arreglos en problemas de gestión de datos en forma crítica y creativa en C++.	Formativa	Rúbrica
Identifica correctamente algoritmos computacionales en procesos de ordenamiento o búsquedas de datos reduciendo costos de programación en la solución de problemas en C++.	Sumativa	Examen
Aplica operaciones de pilas y colas a problemas computacionales, considerando las estructuras de datos dinámicas como estas en soluciones basados en el paradigma de la programación orientada a objetos en C++.	Formativa	Rúbrica
Establece pautas concretas en la definición y utilización de punteros en las soluciones de problemas computacionales, estableciendo criterios y funcionalidades de esta en los programas desarrollados en C++.	Formativa	Rúbrica
Crea programas aplicando punteros basados en la programación orientada a objetos de forma puntual y eficiente en C++.	Formativa	Rúbrica
Utiliza las clases como parte de las soluciones a problemas computacionales especificando los atributos y métodos de cada una de ellas y aplicando los objetos correctamente.	Formativa	Rúbrica
Aplica la herencia como parte fundamental en las clases compuestas dado que se incorpora esquemas del paradigma de la programación orientada a objetos de forma sistemática y estructural.	Formativa	Rúbrica
Incorpora en los códigos de programación las clases de exception para identificar y	Formativa	Rúbrica



solucionar errores de programación de forma puntual y simplificada.		
Establece criterios de selección en el uso de archivos en la solución computacional, identificando la información procesada y su funcionalidad, de forma sistemática y formal.	Formativa	Rúbrica
Propone esquemas de programación para la gestión de datos a través de archivos mediante la organización, tratamiento y mantenimiento de la información en ellos.	Sumativa	Examen

b) Estimaciones no cuantificables

Evaluación	Instrumento	¿Qué evalúa?
Evaluación diagnóstica	Lista de cotejo	Nivel de lógico matemático
Autoevaluación	Lista de cotejo	Dominio del tema
Co-Evaluación	Rúbrica	Funcionalidad de los programas



VII. Administración de los instrumentos y registro de evidencias.

Evaluaciones	Indicador	Evidencias	Instrumento	Puntaje
Primera evaluación parcial	Diseña una solución computacional a problemas planteados a través de estructuras de control selectivas de forma ordenada y sistemática.	Desempeño	Lista de cotejo	5%
	Elabora un esquema algorítmico a través de estructuras de control repetitivas para simplificar y solucionar problemas bajo un esquema computacional de manera puntual y estructurado.	Desempeño	Lista de cotejo	5%
	Verifica a través de pruebas de escritorio las propuestas algorítmicas en la solución de problemas de manera precisa e individual.	Conocimiento	Lista de cotejo	5%
	Soluciona problemas computacionales a través del lenguaje de programación C y C++, incorporando elementos significativos del lenguaje bajo un esquema estructurado y puntual.	Desempeño	Rúbrica	10%
	Desarrolla soluciones computacionales a través de programas en lenguaje C y C++, empleando la estructura completa y funcional del lenguaje.	Producto	Rúbrica	10%
	Diseña funciones específicas en lenguaje C++ para resolver problemas computacionales aplicando la optimización de código y la aplicación de toma de decisiones.	Desempeño	Rúbrica	10%



	Desarrolla funciones avanzadas para incorporar a los programas computacionales de forma sistemática, definidas y puntual en C++	Desempeño	Rúbrica	10%
	Incorpora como solución estructuras de datos estáticas como los arreglos en problemas de gestión de datos en forma crítica y creativa en C++.	Desempeño	Rúbrica	15%
	Identifica correctamente algoritmos computacionales en procesos de ordenamiento o búsquedas de datos reduciendo costos de programación en la solución de problemas en C++.	Conocimiento	Examen	30%
Total				100%
Segunda evaluación parcial	Aplica operaciones de pilas y colas a problemas computacionales, considerando las estructuras de datos dinámicas como éstas en soluciones basados en el paradigma de la programación orientada a objetos en C++.	Desempeño	Rúbrica	10%
	Establece pautas concretas en la definición y utilización de punteros en las soluciones de problemas computacionales, estableciendo criterios y funcionalidades de esta en los programas desarrollados en C++.	Desempeño	Rúbrica	10%
	Crea programas aplicando punteros basados en la programación orientada a	Producto	Rúbrica	10%



	objetos de forma puntual y eficiente en C++.			
	Utiliza las clases como parte de las soluciones a problemas computacionales especificando los atributos y métodos de cada una de ellas y aplicando los objetos correctamente.	Desempeño	Rúbrica	10%
	Aplica la herencia como parte fundamental en las clases compuestas dado que se incorpora esquemas del paradigma de la programación orientada a objetos de forma sistemática y estructural.	Desempeño	Rúbrica	10%
	Incorpora en los códigos de programación las clases de exception para identificar y solucionar errores de programación de forma puntual y simplificada.	Desempeño	Rúbrica	10%
	Establece criterios de selección en el uso de archivos en la solución computacional, identificando la información procesada y su funcionalidad, de forma sistemática y formal.	Conocimiento	Rúbrica	10%
	Propone esquemas de programación para la gestión de datos a través de archivos mediante la organización, tratamiento y mantenimiento de la información en ellos.	Conocimiento	Examen	30%
Total				100%
Evaluación ordinaria	Emplear las herramientas, estructuras y funciones del	Producto	Rúbrica	100%



Evaluación extraordinaria	lenguaje C/C++ y los elementos de la programación estructurada para construir códigos en un lenguaje orientado a objetos que den solución óptima a problemas económicos y financieros.	Producto	Rúbrica	100%
Evaluación a título de suficiencia		Producto	Rúbrica	100%

VIII. Evaluación del aprendizaje.

a) Interpretación de apreciaciones y/o datos.

Para lograr el objetivo de la unidad de aprendizaje, se proponen como instrumentos de evaluación la rúbrica y lista de cotejo como parte de la evaluación formativa, ya que en su mayoría de los entregables serán productos de programación que deben cumplir con ciertas características en cada una de las unidades y estos instrumentos proporcionarán tanto al docente como al alumno pautas y criterios que deberán seguir de manera puntual y precisa tanto en la revisión de los entregables por parte del docente, así como incorporar los criterios a revisar por parte del alumno. Otro de los instrumentos es el examen como parte de una evaluación sumativa permitiendo concretar el conocimiento integral visto en cada una de las unidades.

La naturaleza de estos instrumentos brinda un soporte y ayuda al docente como al alumno como una guía observable del desempeño que este último debe tener en el proceso de enseñanza-aprendizaje de la unidad de aprendizaje. Dentro de la presente guía se permite evidenciar el proceso de adquisición de conocimiento básico para los profesionales de las ciencias administrativas y su aplicación en las diversas áreas donde este perfil profesional tiene injerencia.

b) Juicios y conclusiones valorativas.

Para acreditar la unidad de aprendizaje, el alumno debe cumplir con los lineamientos indicados en el reglamento de Facultades y Escuelas Profesionales y el reglamento interno de la Facultad de Economía, de la Universidad Autónoma del Estado de México, sobre la asistencia y el desempeño académico. Debe, también, cumplir con la entrega de evidencias acordes con el programa de estudios vigente.

También se considerarán como componentes de la evaluación, aspectos referentes al grado de cumplimiento de los objetivos del programa de estudio, mismos que serán valorados por el docente, en función de las habilidades y competencias desarrolladas por los alumnos, las que serán puestas en discusión en los momentos de retroalimentación del curso.

Todo lo anterior tendrá como finalidad intervenir en múltiples niveles, el primero de ellos, asociado con el reconocimiento de destrezas y áreas de oportunidad a nivel



individual, mismas que impactan en perfil de egreso. Por otra parte, los resultados obtenidos en las evaluaciones de carácter cuantitativo y cualitativo tendrán relevancia en la discusión colegiada sobre las estrategias necesarias para cubrir los objetivos planteados en la unidad de aprendizaje. Asimismo, las evaluaciones permitirán a los tutores contar con un indicador sobre el desempeño de los alumnos, que les permita asesorar acerca de las estrategias de aprendizaje necesarias para concretar un aprovechamiento sustantivo.

Por su parte, el alumno deberá ser puntual durante todo el semestre, cubrir el 80% de asistencia para tener derecho a examen ordinario, 60% para extraordinario y 30% para título de suficiencia.

c) Asignación, entrega y revisión de resultados.

La asignación, entrega y revisión de resultados, se hará en apego al calendario escolar correspondiente y al Reglamento de Escuelas y Facultades de la Universidad Autónoma del Estado de México.

El docente tiene la responsabilidad de entregar calificaciones parciales, ordinarias, extraordinarias y a título de suficiencia; expresadas en sistema decimal, en escala de 0 a 10 puntos, en tiempo y forma ante el departamento de control escolar y en el término de cinco días naturales siguientes a la fecha en que se aplique la evaluación; así como dar revisiones a los alumnos a más tardar una sesión posterior a la realización del examen.

Las evaluaciones se llevarán a cabo en los plazos señalados por el Consejo de Gobierno, dentro del período estipulado por el calendario escolar, que se dará a conocer al inicio de cada semestre. Las evaluaciones se efectuarán en los recintos de cada Facultad o Escuelas, dentro de los horarios que al efecto señale la Dirección misma.

Posterior a su evaluación, el alumno podrá solicitar una revisión de calificación en los primeros cinco días naturales posteriores a la fecha de aplicación del examen.