

Programa de estudios por competencias
Algoritmia

1. Identificación del curso

Programa educativo: Ing. En Computación		Unidad de aprendizaje: Algoritmia			Departamento de adscripción: Ciencias Biológicas		
Academia: Sistemas Digitales y de Información		Programa elaborado por: María Obdulia González Fernández, Jesús Salas Ramírez			Modificado por:		Fecha elaboración/modificación: Julio 2013
Clave de la asignatura:	Horas teóricas:	Horas prácticas:	Total de Horas:	Créditos:	Tipo de materia	Área de formación:	Modalidad:
	51	17	68	8	Curso	Básica común	
Conocimientos previos:		Unidad de aprendizaje precedente:			Unidad de aprendizaje subsecuente:		
Razonamiento lógico, nociones de matemáticas básicas y conocimientos básicos del funcionamiento de los sistemas computacionales.		Ninguno			Ninguno		

2. Presentación

La presente unidad de aprendizaje el alumno desarrollará la capacidad de análisis de problemas mediante la creación de algoritmos y pseudocódigo. Es por ello que el estudiante requiere un pensamiento lógico, identificar el proceso de creación de un programa y el desarrollo de algoritmos. Esta asignatura aporta al perfil de egreso del Ingeniero en computación, la competencia de diseñar y desarrollar sistemas computacionales encontrando soluciones creativas a los problemas que le plantee en la vida cotidiana.

Dr. Juan Jorge Rodríguez Bautista
Jefe del departamento

Mtra Claudia Islas T
Presidente de Academia

3. Competencia general (Unidad de competencia)

Genera algoritmos computables pertinentes para el análisis de una situación problemática de su vida cotidiana mediante la aplicación del pensamiento lógico.

4. Elementos de competencia

a. Crea algoritmos básicos secuenciales para el desarrollo de programas computacionales de bajo nivel de dificultad en base a métodos lógicos		
Requisitos		
Cognitivos: (Contenidos).	Procedimentales:	Actitudinales:
<ul style="list-style-type: none"> Define algoritmo, problema, pseudocódigo y lenguaje de programación. Identifica variables, funciones de entrada, salida y operadores "&" y "%". 	<ul style="list-style-type: none"> Examina la interacción entre el código fuente y el resultado en consola. Realiza pruebas de escritorio. Predice el resultado en consola de un algoritmo. Discute diversas formas de creación de un mismo algoritmo. 	<ul style="list-style-type: none"> Acepta y respeta las opiniones de los demás. Colabora con sus compañeros con la finalidad de mejorar el trabajo en equipo. Muestra interés al aprendizaje continuo y autogestivo. Valora la retroalimentación grupal.
Estrategias didácticas:	Recursos requeridos	Tiempo estimado:
Conferencia magistral Estudio de casos Solución de problemas Organizadores gráficos	-Bibliografía básica -Cañón, laptop, plataforma virtual	4 semanas
Criterios de desempeño:	Evidencias:	Producto esperado:
Genera pseudocódigo de manera lógica y ordenada. Demuestra mediante pruebas de escritorio el correcto funcionamiento de un algoritmo. Argumenta la pertinencia de sus algoritmos.	Creación de diversos algoritmos convertidos a pseudocódigo con sus respectivos diagramas de flujo.	Genera un documento en el que se explica el análisis de un problema que se resuelve mediante la creación de un algoritmo.

Transfiere un algoritmo a pseudocódigo y construye un diagrama de flujo correctamente.		
--	--	--

b- Crea algoritmos a partir de estructuras de control selectivas y repetitivas para el desarrollo de programas computacionales de bajo nivel de dificultad mediante razonamiento lógico.		
Requisitos		
Cognitivos: (Contenidos).	Procedimentales:	Actitudinales:
<ul style="list-style-type: none"> Explica las estructuras básicas de control secuenciales selectivas y de repetición. 	<ul style="list-style-type: none"> Diagrama de forma lógica un problema que implique estructuras de secuencia, decisión y repetición. Crea pseudocódigos que resuelva problemas que requieran estructuras de secuencia, decisión y repetición. 	<ul style="list-style-type: none"> Acepta y respeta las opiniones de sus compañeros. Colabora con sus compañeros con la finalidad de mejorar el trabajo en equipo. Muestra interés al aprendizaje continuo y autogestivo. Valora la retroalimentación grupal.
Estrategias didácticas:	Recursos requeridos	Tiempo estimado:
-Investigación documental -estudio de casos, solución de problemas -Conferencia magistral -Organizadores gráficos.	-Bibliografía básica -Cañón, laptop, plataforma virtual	4 semanas
Criterios de desempeño:	Evidencias:	Producto esperado:
Genera pseudocódigo de manera lógica y ordenada. Demuestra mediante pruebas de escritorio el correcto funcionamiento de un algoritmo. Argumenta la pertinencia de sus algoritmos.	-Reporte de investigación -Cuadros de doble entrada de estructuras de control y lenguajes de programación. -Diagramas de flujo -ejercicios de pseudocódigos	Reporte que contenga el planteamiento del problema, el diagrama de flujo y el pseudocódigo del estudio de caso propuesto.

Transfiere un algoritmo a pseudocódigo y construye un diagrama de flujo correctamente.		
--	--	--

c. Crea algoritmos para el desarrollo de programas computacionales de bajo nivel de dificultad estructurándolos de forma modular		
Requisitos		
Cognitivos: (Contenidos).	Procedimentales:	Actitudinales:
<ul style="list-style-type: none"> Identifica el concepto de modulo dentro de un algoritmo. Describe la secuencia de un algoritmo estructurado de forma modular. Identifica los parámetros por valor y por referencia de un algoritmo estructurado de forma modular Examina la optimización de un algoritmo mediante el uso de módulos con pasos de parámetros. 	<ul style="list-style-type: none"> Diagrama de forma lógica un problema que implique algoritmos modulares con paso de parámetros. Crea pseudocódigos que resuelva problemas que requieran la modularización del código fuente. Predice el comportamiento de un algoritmo estructurado en módulos mediante pruebas de escritorio. 	<ul style="list-style-type: none"> Acepta y respeta las opiniones de sus compañeros. Colabora con sus compañeros con la finalidad de mejorar el trabajo en equipo. Muestra interés al aprendizaje continuo y autogestivo. Valora la retroalimentación grupal.
Estrategias didácticas:	Recursos requeridos	Tiempo estimado:
-Investigación documental -estudio de casos, solución de problemas -Conferencia magistral -Organizadores gráficos.	-Bibliografía básica -Cañón, laptop, plataforma virtual	3 semanas
Criterios de desempeño:	Evidencias:	Producto esperado:
Genera pseudocódigo de manera lógica y ordenada. Demuestra mediante pruebas de escritorio el correcto funcionamiento de un algoritmo. Argumenta la pertinencia de sus algoritmos.	-Reporte de investigación -Mapa mental de los módulos que integra un algoritmo -Diagramas de flujo -ejercicios de pseudocódigos	-Informe de la creación de un juego con números al azar solventado mediante el uso de programación modular .

d. Evalúa algoritmos de bajo nivel para establecer los recursos consumidos haciendo uso de técnicas preestablecidas		
Requisitos		
Cognitivos: (Contenidos).	Procedimentales:	Actitudinales:
<ul style="list-style-type: none"> • Identifica los recursos que consume cada elemento de un algoritmo. • Describe los métodos para evaluar el costo de un algoritmo. 	<ul style="list-style-type: none"> • Compara el consumo de recursos entre dos algoritmos similares. • Establece la eficiencia de algoritmos de bajo nivel (selectivo, repetitivo y modular). • Mide el consumo de recursos de algoritmos de bajo nivel (selectivo, repetitivo y modular). 	<ul style="list-style-type: none"> • Acepta y respeta las opiniones de sus compañeros. • Colabora con sus compañeros con la finalidad de mejorar el trabajo en equipo. • Muestra interés al aprendizaje continuo y autogestivo. • Valora la retroalimentación grupal.
Estrategias didácticas:	Recursos requeridos	Tiempo estimado:
-Investigación documental -estudio de casos de forma colaborativa -Conferencia magistral -Organizadores gráficos.	-Bibliografía básica -Cañón, laptop, plataforma virtual	3 semanas
Criterios de desempeño:	Evidencias:	Producto esperado:
Emplea de forma correcta los métodos para evaluar los recursos consumidos en un algoritmo.	-Reporte de investigación -Reporte de evaluación colaborativa de algoritmos	-Informe sobre la evaluación de un algoritmo.

Se agregaran tantos cuadros sean necesarios dependiendo de los elementos de competencia que se han identificado.

c- Evaluación y acreditación

Dr. Juan Jorge Rodríguez Bautista
 Jefe del departamento

Mtra Claudia Islas T
 Presidente de Academia

Área de conocimiento:

- a) Examen departamental 20%
- b) actividades de aprendizaje (Tareas) 30%

Área de habilidades y destrezas:

- a) Actividades integradoras 40%

Área de actitud:

- a) Participación 5%
- b) Coevaluación 5%

(Podrá agregar cuantos elementos requiera en cada uno de los apartados de evaluación)

d- Bibliografía

Sznajdleder P. (2011) Algoritmo a fondo con implementaciones en C y Java. Alfa Omega, 1 edición México. pp553

Lee R. (2007) Introducción al diseño y análisis de algoritmos. MC Graw Hil 1 edición, México, pp736

López G. (2009) Análisis y Diseño de Algoritmos, Alfa Omega, 1 edición, México .

e- Perfil docente

El docente de esta materia deberá ser un profesionalista con formación en las áreas de la computación, comunicaciones o informática; capaz de motivar a la investigación y creación de conocimiento, con habilidades para transmitir sus conocimientos y enseñar de forma interactiva propiciando en los alumnos el auto-aprendizaje.



Dr. Juan Jorge Rodríguez Bautista
Jefe del departamento



Mtra Claudia Islas T
Presidente de Academia