



Universidad Nacional Autónoma de México
Facultad de Contaduría y Administración
Sistema Universidad Abierta y Educación a Distancia

Licenciatura en Informática

Análisis Diseño e Implantación de Algoritmos

Cuaderno de actividades



COLABORADORES

COORDINACIÓN GENERAL

Mtra. Gabriela Montero Montiel
Jefe de la División SUAyED-FCA-UNAM

COORDINACIÓN ACADÉMICA

Mtro. Francisco Hernández Mendoza
FCA-UNAM

COAUTORES

L.C. Gilberto Manzano Peñaloza
Mtro. Luis Fernando Zúñiga Hernández

ACTUALIZACIÓN

Mtro. René Montesano Brand

DISEÑO INSTRUCCIONAL

Lic. Chantal Ramírez Pérez
Mayra Lilia Velasco Chacón

CORRECCIÓN DE ESTILO

L.F. Francisco Vladimir Aceves Gaytán

DISEÑO DE PORTADAS

L.CG. Ricardo Alberto Báez Caballero
Mtra. Marlene Olga Ramírez Chavero
L.DP. Ethel Alejandra Butrón Gutiérrez

DISEÑO EDITORIAL

Mtra. Marlene Olga Ramírez Chavero

Contenido

Datos de identificación	5
Sugerencias de apoyo	6
Instrucciones para trabajar con el cuaderno de actividades	7
Objetivo general de la asignatura y Temario Oficial	9
Unidad 1. Fundamentos de algoritmos	10
Objetivo particular y temario detallado	11
Actividad diagnóstica	12
Actividades de aprendizaje	13
Actividad integradora	15
Cuestionario de reforzamiento	16
Examen parcial de la unidad (Autoevaluación)	17
Respuestas	20
Unidad 2. Análisis de algoritmos	21
Objetivo particular y temario detallado	22
Actividad diagnóstica	23
Actividades de aprendizaje	24
Actividad integradora	25
Cuestionario de reforzamiento	26
Examen parcial de la unidad (Autoevaluación)	27
Respuestas	29
Unidad 3. Diseño de algoritmos para la solución de problemas	30
Objetivo particular y temario detallado	31
Actividad diagnóstica	32
Actividades de aprendizaje	33
Actividad integradora	34
Cuestionario de reforzamiento	35
Examen parcial de la unidad (Autoevaluación)	36
Respuestas	39

Unidad 4. Implantación de algoritmos	40
Objetivo particular y temario detallado	41
Actividad diagnóstica	42
Actividades de aprendizaje	43
Actividad integradora	44
Cuestionario de reforzamiento	45
Examen parcial de la unidad (de autoevaluación)	46
Respuestas	49
Unidad 5. Evaluación de algoritmos	50
Objetivo particular y temario detallado	51
Actividad diagnóstica	52
Actividades de aprendizaje	53
Actividad integradora	54
Cuestionario de reforzamiento	55
Examen parcial de la unidad (de autoevaluación)	56
Respuestas	59

DATOS DE IDENTIFICACIÓN

Análisis diseño e Implantación de Algoritmos	Clave: 1164
Plan: 2012	Créditos: 8
Licenciatura: Informática	Semestre: 1°
Área o campo de conocimiento: Matemáticas	Horas por semana: 4
Duración del programa: semestral	Requisitos: ninguno
Tipo: Teórica Teoría: 4 Práctica: 0	
Carácter: Obligatoria (<input checked="" type="checkbox"/>) Optativa (<input type="checkbox"/>)	
Seriación: Si (<input type="checkbox"/>) No (<input checked="" type="checkbox"/>) Obligatoria (<input type="checkbox"/>) Indicativa (<input type="checkbox"/>)	
Asignatura con seriación antecedente: Ninguna	
Asignatura con seriación subsecuente: Estadística inferencial	

SUGERENCIAS DE APOYO

- Trata de compartir tus experiencias y comentarios sobre la asignatura con tus compañeros, a fin de formar grupos de estudio presenciales o a distancia (comunidades virtuales de aprendizaje, a través de foros de discusión y correo electrónico, etcétera), y puedan apoyarse entre sí.
- Programa un horario propicio para estudiar, en el que te encuentres menos cansado, ello facilitará tu aprendizaje.
- Dispón de periodos extensos para al estudio, con tiempos breves de descanso por lo menos entre cada hora si lo consideras necesario.
- Busca espacios adecuados donde puedas concentrarte y aprovechar al máximo el tiempo de estudio.

Instrucciones para trabajar con el cuaderno de actividades

El programa de la asignatura consta de 5 unidades. Por cada unidad encontrarás una serie de actividades. El número de las mismas varía de acuerdo a la extensión de la unidad.

Notarás que casi todas las unidades comienzan con la elaboración de un mapa conceptual ó mental; esto es con el fin de que tu primera actividad sea esquematizar el contenido total de la unidad para que tenga una mejor comprensión y dominio total de los temas.

Te recomendamos que leas detenidamente cada actividad a fin de que te quede claro qué es lo que tienes que realizar. Si al momento de hacerlo algo no queda claro, no dudes en solicitar el apoyo de tu asesor quien te indicará la mejor forma de realizar tu actividad en asesorías semipresenciales o por correo electrónico para los alumnos de la modalidad abierta, o bien para la modalidad a distancia a través de los medios proporcionados por la plataforma.

Te sugerimos (salvo la mejor opinión de tu asesor), seguir el orden de las unidades y actividades, pues ambas están organizadas para que tu aprendizaje sea gradual. En el caso de los alumnos de la modalidad a distancia, la entrega de actividades está sujeta al plan de trabajo establecido por cada asesor y el trabajo es directamente en plataforma educativa:

<http://fcaenlinea1.unam.mx/licenciaturas/>

La forma en que deberás responder a cada actividad dependerá de la instrucción dada (número de cuartillas, formatos, si hay que esquematizar, etcétera).

Una vez que hayas concluido las actividades entrégalas a tu asesor si, así él te lo solicita. Los alumnos de la modalidad a distancia, deberán realizar la actividad directamente en la plataforma educativa de acuerdo a la instrucción dada.

Te invitamos a que trabajes estas actividades con el mayor entusiasmo, pues fueron elaboradas considerando apoyarte en tu aprendizaje de ésta asignatura.



Indicaciones:

Notarás que tanto los cuestionarios de reforzamiento como las actividades de aprendizaje, contienen instrucciones tales como “adjuntar archivo”, “trabajo en foro”, “texto en línea”, “trabajo en wiki o en Blog”, indicaciones que aplican específicamente para los estudiantes del SUAYED de la modalidad a distancia. Los alumnos de la modalidad abierta, trabajarán las actividades de acuerdo a lo establecido por el asesor de la asignatura en su plan de trabajo, incluyendo lo que sé y lo que aprendí.



Biblioteca Digital:

Para tener acceso a otros materiales como libros electrónicos, es necesario que te des de alta a la Biblioteca Digital de la UNAM (BIDI). Puedes hacerlo desde la página principal de la FCA <http://www.fca.unam.mx/> **Alumnos, >Biblioteca >Biblioteca digital >Clave para acceso remoto >Solicita tu cuenta.** Elige la opción de “Alumno” y llena los campos solicitados. Desde este sitio, también puedes tener acceso a los libros electrónicos.

OBJETIVO GENERAL

Al finalizar el curso, el alumno será capaz de implementar algoritmos en un lenguaje de programación.

TEMARIO OFICIAL

(64 HORAS)

	Horas
1. Fundamentos de algoritmos	12
2. Análisis de algoritmos	12
3. Diseño de algoritmos para la resolución de problemas	12
4. Implantación de algoritmos	12
5. Evaluación de algoritmos	16
Total	64

UNIDAD 1

Fundamentos de algoritmos

OBJETIVO PARTICULAR

Al finalizar la unidad, el alumno podrá identificar los componentes y propiedades de los algoritmos.

TEMARIO DETALLADO

(12 HORAS)

1. Fundamentos de algoritmos

1.1 Definición de algoritmo

1.2 Propiedades de los algoritmos

1.3 Autómatas y lenguajes formales

1.4 Máquina de Turing

ACTIVIDAD DIAGNÓSTICA

LO QUE SÉ



Texto en línea.

Con base en tus conocimientos y con tus propias palabras, redacta una definición de algoritmo.

ACTIVIDADES DE APRENDIZAJE



Unidad 1, actividad inicial. *Adjuntar archivo.* A partir del estudio de la bibliografía específica sugerida, elabora un mapa conceptual u [organizador gráfico](#) con los temas de la unidad. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

1. **Unidad 1, actividad 1. *Adjuntar archivo.*** Investiga tres ejemplos de aplicación de un algoritmo en diferentes campos e identifica sus características según lo analizado en esta unidad.
2. **Unidad 1, actividad 2. *Adjuntar archivo.*** Elabora un algoritmo de ordenamiento de datos donde se genere un intercambio de posición de los números 24 y 9, dando como salida 9 y 24 en este orden. Te sugerimos utilizar tres variables: num1, num2 y aux, para que puedas realizar el intercambio. Representa el algoritmo en forma de pasos sucesivos (tipo receta de cocina) empleando un lenguaje natural; evita el lenguaje de programación y el diagrama de flujo.
3. **Unidad 1, actividad 3. *Adjuntar archivo.*** Realiza una investigación en libros, revistas, Internet, etc. de los conceptos presentados a continuación. Con base en lo investigado, da un ejemplo de situaciones en las que se pueden aplicar las siguientes demostraciones:
 - Demostraciones deductivas
 - Demostraciones de la conversión contradictoria
 - Demostración por deducción del absurdo
 - Contraejemplos
 - Demostraciones inductivas
 - Inducciones estructurales

4. **Unidad 1, actividad 4. *Adjuntar archivo.*** Investiga un ejemplo del empleo de un autómata y otro de una máquina de Turing; e identifica las características de los algoritmos aplicados en ellos. Presenta los ejemplos y escribe tus observaciones.
5. **Unidad 1, actividad 5. *Adjuntar archivo.*** Una vez comprendido el tema de máquina de Turing, identifica las características de un lenguaje decidable, aceptable, semidecidible, indecidible. Apóyate de una tabla comparativa.

ACTIVIDAD INTEGRADORA

LO QUE APRENDÍ



Adjuntar archivo.

Retoma la definición de algoritmo que anotaste en el apartado “Lo que sé” y complementácela con lo estudiado en la unidad y otras fuentes (en este caso, no olvides citarlas). Presenta tu definición ampliada.

CUESTIONARIO DE REFORZAMIENTO



Adjuntar archivo. Responde las siguientes preguntas.

1. ¿Qué es un algoritmo?
2. ¿Cuáles son las características de un algoritmo?
3. ¿Qué es un autómata?
4. Explica por qué un termostato puede ser considerado un autómata.
5. ¿Qué es un diagrama de estado?
6. ¿Qué es una tabla de estado?
7. En el campo de los autómatas, ¿qué es un alfabeto?
8. ¿Qué es una cadena vacía?
9. ¿Cuál es la definición de lenguaje?
10. ¿Qué es y para qué sirve una gramática?
11. Da un ejemplo de una regla de producción BNF.
12. ¿Qué elementos constituyen un lenguaje formal?
13. Describe brevemente tres tipos de gramáticas.
14. ¿Qué es una máquina de Turing?
15. ¿Qué es un proceso computable?

EXAMEN PARCIAL

(de autoevaluación)



I. Responde verdadero (V) o falso (F).

	V	F
1. Es muy factible obviar pasos que sean repetitivos en las operaciones que realiza una computadora.	<input type="radio"/>	<input type="radio"/>
2. La posibilidad de aplicar diversos criterios es una característica de los algoritmos.	<input type="radio"/>	<input type="radio"/>

II. Marca en el cuadro el concepto correspondiente a cada definición.

<p>___ 1. Conjunto de todos los símbolos válidos o posibles para una aplicación.</p>	<p>a) Frase b) Gramática c) Alfabeto d) Lenguaje e) Cadena vacía</p>
<p>___ 2. Es la asociación de un conjunto de símbolos definidos en un alfabeto (cadena), con la propiedad de tener sentido, significado y lógica.</p>	
<p>___ 3. Conjunto de cadenas que obedecen a un alfabeto fijado.</p>	

<p>___ 4. La longitud del conjunto de caracteres que utiliza es igual a cero.</p> <p>___ 5. Colección estructurada de palabras y frases ligadas por reglas que definen el conjunto de cadenas de caracteres que representan los comandos completos y pueden ser reconocidos por un motor de discurso.</p>	
---	--

III. Responde verdadero (V) o falso (F).

	V	F
1. Los procesos computables y no computables pueden implementarse en un algoritmo o máquina de Turing.	<input type="radio"/>	<input type="radio"/>
2. A un proceso no computable se le puede generar un lenguaje decible para que sea leído en una máquina de Turing.	<input type="radio"/>	<input type="radio"/>
3. La cinta de una máquina de Turing es necesariamente finita, ya que está definida por el tamaño del programa.	<input type="radio"/>	<input type="radio"/>
4. El programa es un conjunto de instrucciones que controla los movimientos de la cabeza de lectura-escritura.	<input type="radio"/>	<input type="radio"/>
5. La cinta es un dispositivo sólo de lectura y no de almacenamiento.	<input type="radio"/>	<input type="radio"/>

IV. Selecciona la respuesta correcta.

1. Es una característica de un algoritmo.

- | | |
|---|---|
| <input type="radio"/> a) Acepta criterios en su desarrollo. | <input type="radio"/> b) Puede omitir pasos. |
| <input type="radio"/> c) En ocasiones, no obtiene un resultado. | <input type="radio"/> d) Contiene una condición que detiene su ejecución. |

2. Inventor del Pato con Aparato Digestivo.

- | | |
|---|---|
| <input type="radio"/> a) Pierre Jacquet Drozz | <input type="radio"/> b) Falcon |
| <input type="radio"/> c) Josheph Marie Jacquard | <input type="radio"/> d) Jacques Vacanson |

3. Autómata formado por cinta, cabeza de lectura-escritura y programa.

- | | |
|--|--|
| <input type="radio"/> a) Máquina de Turing | <input type="radio"/> b) El Dibujante |
| <input type="radio"/> c) Los Músicos | <input type="radio"/> d) El Telar Automático |

4. Problema que no puede implementarse en una máquina de Turing por no tener solución para todas sus posibles entradas.

- | | |
|-------------------------------------|---------------------------------------|
| <input type="radio"/> a) Computable | <input type="radio"/> b) Indecidible |
| <input type="radio"/> c) Decidible | <input type="radio"/> d) Disfuncional |

5. Tipo de gramática en la que el lado derecho de la regla de producción siempre debe ser igual o mayor que el lado izquierdo.

- | | |
|--|--|
| <input type="radio"/> a) Independientes del contexto | <input type="radio"/> b) Sensible al contexto |
| <input type="radio"/> c) Regulares | <input type="radio"/> d) Ninguna de las anteriores |

RESPUESTAS

EXAMEN DE AUTOEVALUACIÓN



En este apartado encontrarás las respuestas al examen por unidad.

Unidad 1
I. Solución
1. F
2. F

Unidad 1
II. Solución
1. c
2. a
3. e
4. d
5. b

Unidad 1
III. Solución
1. F
2. F
3. F
4. V
5. F

Unidad 1
IV. Solución
1. d
2. d
3. a
4. b
5. b

Análisis de algoritmos

OBJETIVO PARTICULAR

Al finalizar la unidad, el alumno podrá analizar un problema determinado y buscar una solución a partir de un algoritmo.

TEMARIO DETALLADO

(12 HORAS)

2.1. Análisis de algoritmos

2.1.1 Análisis del problema

2.1.2 Computabilidad

2.1.3 Algoritmos cotidianos

2.1.4 Algoritmos recursivos

2.1.5 Algoritmos de búsqueda y ordenación

ACTIVIDAD DIAGNÓSTICA

LO QUE SÉ



Adjuntar archivo.

Investiga los elementos de un problema y relaciónalos con los algoritmos. Anota tus conclusiones.

ACTIVIDADES DE APRENDIZAJE



Unidad 2, actividad inicial. Adjuntar archivo. A partir del estudio de la bibliografía específica sugerida, elabora un mapa conceptual u [organizador gráfico](#) con los temas de la unidad. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

- Unidad 2, actividad 1. Actividad en foro.** Investiga cinco ejemplos de problemas decidibles y no decidibles y coméntalos con tu asesor en el foro de la asignatura.
Si tu asignatura la trabajas fuera de plataforma educativa, entonces realiza la misma actividad en no más de una cuartilla y entrégala a tu asesor.
- Unidad 2, actividad 2. Actividad en foro.** Investiga las diferencias entre la solución iterativa y la solución recursiva y coméntalas en el foro de la asignatura.
Si tu asignatura la trabajas fuera de plataforma educativa, entonces realiza la misma actividad en no más de una cuartilla y entrégala a tu asesor.
- Unidad 2, actividad 3. Adjuntar archivo.** Realiza un cuadro comparativo con las características de los métodos de ordenación: burbuja, inserción, selección, *quick sort* y *shell* e identifica sus diferencias y su aplicación.
- Unidad 2, actividad 4. Adjuntar archivo.** Investiga un ejemplo donde se aplique una función hash multiplicativa y otra por división. Con base en esos ejemplos, menciona a grandes rasgos la utilidad que tienen estas funciones y la forma de implementarlas a través de un algoritmo.
- Unidad 2, actividad 5. Adjuntar archivo.** Elabora un ejemplo de un problema donde emplees cualquiera de los métodos de ordenamiento estudiados en esta unidad para su solución. Elabora el algoritmo correspondiente en un documento de Word.

ACTIVIDAD INTEGRADORA

LO QUE APRENDÍ



Adjuntar archivo.

Retoma la actividad del apartado “Lo que sé” y complementala con lo estudiado en esta unidad y la información de otras fuentes (no olvides citarlas).

CUESTIONARIO DE REFORZAMIENTO



Adjuntar archivo. Responde las siguientes preguntas.

1. ¿Qué elementos se deben considerar para determinar el rendimiento de un algoritmo?
2. ¿Qué factores podrían influir en forma negativa para precisar con exactitud el rendimiento de los algoritmos?
3. ¿Qué es un modelo?
4. ¿Qué son los problemas decidibles?
5. ¿Qué entiendes por recursividad?
6. ¿Qué entiendes por inducción?
7. Describe el método para calcular la complejidad de una función recursiva.
8. ¿Cuál es el método de ordenación menos eficiente y cuál el más eficiente?
9. Explica el concepto divide y vencerás que utiliza el método de ordenación *quick sort*.
10. ¿Qué diferencia existe entre una tabla *hash* y una función *hash*? ¿En qué consisten ambas?

EXAMEN PARCIAL

(de autoevaluación)



I. Responde verdadero (V) o falso (F).

	V	F
1. Un problema puede llegar a tener más de un algoritmo que lo solucione.	<input type="radio"/>	<input type="radio"/>
2. La cantidad de datos de entrada y las operaciones determinan el tiempo de ejecución.	<input type="radio"/>	<input type="radio"/>
3. Mediante el empleo de fórmulas matemáticas, es posible conocer el rendimiento de un algoritmo.	<input type="radio"/>	<input type="radio"/>
4. El tiempo de ejecución depende del tipo de datos de salida.	<input type="radio"/>	<input type="radio"/>
5. Un algoritmo se selecciona en función de su tamaño.	<input type="radio"/>	<input type="radio"/>

II. Marca en el cuadro el concepto correspondiente a cada definición.

<p>_____ 1. Propiedad que tienen ciertos problemas de poder resolverse a través de un algoritmo.</p> <p>_____ 2. Es aquel cuya respuesta puede mapearse al conjunto de valores {0,1}.</p> <p>_____ 3. Lenguaje que se implementa para resolver un problema con un número finito de pasos por algoritmo.</p> <p>_____ 4. Tipo de problema cuyo lenguaje no puede ser reconocido por una máquina de Turing.</p> <p>_____ 5. Problemas que pueden resolverse con una máquina de Turing.</p>	<p>a) Problema de decisión</p> <p>b) Computabilidad</p> <p>c) Indecidible</p> <p>d) Recursivo</p> <p>e) Computables</p>
--	---

III. Marca en el cuadro el concepto correspondiente a cada definición.

<p>___ 1. Intercambia elementos que están muy distantes.</p> <p>___ 2. Emplea la técnica divide y vencerás para separar el problema en subproblemas más pequeños.</p> <p>___ 3. Se basa en seleccionar el elemento más pequeño del arreglo y colocarlo en la posición más baja del mismo.</p> <p>___ 4. Es el método más sencillo, pero menos eficiente.</p> <p>___ 5. Método que consiste en tomar un elemento y colocarlo en la posición ordenada correspondiente</p>	<p>a) Burbuja</p> <p>b) Selección</p> <p>c) <i>Shell</i></p> <p>d) Inserción</p> <p>e) <i>Quick-sort</i></p>
---	--

RESPUESTAS

EXAMEN DE AUTOEVALUACIÓN



En este apartado encontrarás las respuestas al examen por unidad.

Unidad 2
I. Solución
1. V
2. F
3. V
4. F
5. F

Unidad 2
II. Solución
1. b
2. a
3. d
4. c
5. e

Unidad 2
III. Solución
1. b
2. e
3. c
4. a
5. d

**Diseño de algoritmos para la
solución de problemas**

OBJETIVO PARTICULAR

Al terminar la unidad, el alumno podrá plantear, desarrollar y seleccionar un algoritmo determinado para solucionar un problema específico.

TEMARIO DETALLADO (12 HORAS)

3. Diseño de algoritmos para la solución de problemas

3.1 Niveles de abstracción para la construcción de algoritmos

3.2 Técnicas de diseño de algoritmos

3.3 Alternativas de solución

3.4 Diagramas de flujo

ACTIVIDAD DIAGNÓSTICA

LO QUE SÉ



Actividad en foro.

Redacta con tus propias palabras que entiendes por abstracción. Una vez hecho la anterior sube tu definición al foro de la asignatura y compártelo con tus compañeros.

Si tu asignatura la trabajas fuera de plataforma educativa, entonces realiza la misma actividad en no más de una cuartilla y entrégala a tu asesor.

ACTIVIDADES DE APRENDIZAJE



Unidad 3, actividad inicial. Adjuntar archivo. A partir del estudio de la bibliografía específica sugerida, elabora un mapa conceptual u [organizador gráfico](#) con los temas de la unidad. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

1. **Unidad 3, actividad 1. Adjuntar archivo.** Elabora un mapa conceptual del contenido de la unidad.
2. **Unidad 3, actividad 2. Adjuntar archivo.** Diseña un algoritmo para dar solución a un problema que propongas, en donde se utilice alguna de las estructuras de control: *MIENTRAS*, *HASTA QUE*, *SI ENTONCES SI NO* y el contador *PARA*.
3. **Unidad 3, actividad 3. Adjuntar archivo.** Elabora un cuadro comparativo de las características de las diferentes técnicas de diseño de algoritmos y determina el tipo de problemas que se puede resolver con cada una.
4. **Unidad 3, actividad 4. Adjuntar archivo.** Diseña un algoritmo voraz para solucionar el problema de dar cambio de dinero por la venta de diversos artículos en una tiendita.
5. **Unidad 3, actividad 5. Adjuntar archivo.** Investiga sobre cómo resolver el juego de las torres de Hanoi y diseña un algoritmo empleando las funciones recursivas necesarias para su ejecución.

ACTIVIDAD INTEGRADORA

LO QUE APRENDÍ



Adjuntar archivo.

A partir de la revisión general de esta unidad y la bibliografía específica sugerida, elabora un mapa mental u [organizador gráfico](#) que te facilite el estudio y comprensión de los temas analizados. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

CUESTIONARIO DE REFORZAMIENTO



Adjuntar archivo. Responde las siguientes preguntas.

1. ¿Cuáles son las estructuras de ciclos?
2. ¿Qué diferencias hay entre las estructuras *MIENTRAS* y *HASTA QUÉ*?
3. Dentro de una estructura *FOR* se puede utilizar una instrucción para cambiar el valor de la variable que emplea la estructura para controlar las iteraciones. Indica por qué no debería cambiarse el valor a esta variable dentro de la misma estructura.
4. ¿Para qué tipo de problemas se utilizan los algoritmos voraces?
5. ¿Qué funciones utiliza un algoritmo voraz?
6. Explica el concepto de recursividad en la técnica divide y vencerás.
7. En programación dinámica, ¿qué se entiende por subestructura óptima?
8. ¿Qué estrategia de diseño está relacionada con la búsqueda combinatoria?
9. ¿Qué tareas realizan los algoritmos *backtracking* cuando encuentran una solución candidata?
10. En un tablero de ajedrez de 8 x 8 casillas, la pieza denominada reina puede avanzar una o varias casillas en forma horizontal, vertical o diagonal. Si en su camino encuentra una pieza adversaria, la ataca. Entonces:
 - ¿Cómo colocarías ocho reinas sobre el tablero sin que alguna reina ataque a la otra?
 - ¿Cuál estrategia de diseño de algoritmos recomendarías para solucionar el problema de las ocho reinas?

EXAMEN PARCIAL

(de autoevaluación)



I. Responde verdadero (V) o falso (F).

	V	F
1. Las condiciones son aquellas que hacen llamadas a sí mismas en su definición, simplificando los valores originales de entrada.	<input type="radio"/>	<input type="radio"/>
2. Los acumuladores se implementan en los casos en que el problema que se desea resolver puede simplificarse en versiones más pequeñas del mismo.	<input type="radio"/>	<input type="radio"/>
3. Los contadores se caracterizan por iterar instrucciones en función de una condición que debe cumplirse en un momento conocido.	<input type="radio"/>	<input type="radio"/>
4. Los ciclos son estructuras que se caracterizan por iterar instrucciones en función de una condición que debe cumplirse en un momento bien definido.	<input type="radio"/>	<input type="radio"/>
5. Las rutinas recursivas regularmente contienen una cláusula condicional.	<input type="radio"/>	<input type="radio"/>

II. Selecciona el inciso que completa correctamente el enunciado.

<p>1. Divide el problema en subproblemas que resuelve recursivamente para, finalmente, reunir las soluciones individuales. _____</p> <p>2. Resuelve el problema en subconjuntos a través de subestructuras óptimas. _____</p> <p>3. Técnica mediante la cual se encuentra la mejor combinación en un momento determinado (búsqueda en profundidad). _____</p> <p>4. Se utilizan en solución de problemas de optimización, aunque son poco eficientes. _____</p>	<p>a) Programación dinámica</p> <p>b) Algoritmos voraces</p> <p>c) Vuelta atrás</p> <p>d) Divide y vencerás</p>
---	---

III. Selecciona la respuesta correcta.

1. Es característico de la estructura HASTA QUÉ:

- | | |
|--|--|
| <input type="radio"/> a) Evalúa una condición al principio de la estructura. | <input type="radio"/> b) Ejecuta las instrucciones y luego evalúa la condición. |
| <input type="radio"/> c) Si la condición evaluada resulta falsa, se sale de la estructura. | <input type="radio"/> d) La estructura al final no está delimitada por un comando. |

2. Una variable del tipo acumulador es aquella que:

- | | |
|--|---|
| <input type="radio"/> a) Se incrementa en cada iteración con la unidad. | <input type="radio"/> b) No sufre un cremento alguno, sólo es de control. |
| <input type="radio"/> c) Aumenta su valor con el valor propio más el del incremento. | <input type="radio"/> d) No se relaciona con la solución arrojada por el algoritmo. |

3. Cuando se sabe con exactitud el número de iteraciones que debe realizar una estructura, se utiliza:

- | | |
|------------------------------------|---|
| <input type="radio"/> a) Para | <input type="radio"/> b) Mientras |
| <input type="radio"/> c) Hasta que | <input type="radio"/> d) Si entonces no |

4. Estructura en la que se puede prescindir del conjunto de instrucciones de la condición falsa.

- | | |
|------------------------------------|---|
| <input type="radio"/> a) Para | <input type="radio"/> b) Mientras |
| <input type="radio"/> c) Hasta que | <input type="radio"/> d) Si entonces no |

5. Técnica de diseño de algoritmos que contiene una función de factibilidad, una de selección y una objetivo.

- | | |
|--|--|
| <input type="radio"/> a) Algoritmos voraces | <input type="radio"/> b) Divide y vencerás |
| <input type="radio"/> c) Programación dinámica | <input type="radio"/> d) Vuelta atrás |

6. Si una rutina posee dos llamadas recursivas, se denominan algoritmos:

- | | |
|--|--|
| <input type="radio"/> a) Voraces | <input type="radio"/> b) Divide y vencerás |
| <input type="radio"/> c) Programación dinámica | <input type="radio"/> d) Vuelta atrás |

RESPUESTAS

(Examen de Autoevaluación)



En este apartado encontrarás las respuestas al examen por unidad.

Unidad 3	
I. Solución	
1.	F
2.	F
3.	V
4.	V
5.	V

Unidad 3	
II. Solución	
1.	d
2.	a
3.	c
4.	b

Unidad 3	
III. Solución	
1.	b
2.	c
3.	a
4.	d
5.	a
6.	b

Implantación de algoritmos

OBJETIVO PARTICULAR

Al finalizar la unidad, el alumno podrá llevar a cabo la realización de un programa a partir de un algoritmo para un problema determinado.

TEMARIO DETALLADO

(12 HORAS)

4.1. El programa como una expresión computable del algoritmo

4.2 Programación estructurada

4.3 Modularidad

4.4 Funciones, rutinas y procedimientos

4.5 Enfoque de algoritmos

ACTIVIDAD DIAGNÓSTICA

LO QUE SÉ



Actividad en foro.

Busca tres definiciones de programación de 3 libros que elijas, subraya las coincidencias de éstas y compártelo en el foro de la asignatura. Trata con tus compañeros de realizar una definición en grupo.

Si tu asignatura la trabajas fuera de plataforma educativa, entonces realiza la misma actividad en no más de una cuartilla y entrégala a tu asesor.

ACTIVIDADES DE APRENDIZAJE



Unidad 4, actividad inicial. Adjuntar archivo. A partir del estudio de la bibliografía específica sugerida, elabora un mapa conceptual u [organizador gráfico](#) con los temas de la unidad. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

1. **Unidad 4, actividad 1. Adjuntar archivo.** Realiza un diagrama de flujo donde muestres el proceso para convertir un algoritmo en un programa de computadora.
2. **Unidad 4, actividad 2. Adjuntar archivo.** Investiga qué otras estructuras de control se derivan de las básicas que se abordaron en este apunte.
3. **Unidad 4, actividad 3. Adjuntar archivo.** Desarrolla un diagrama *top-down* y uno de *bottom-up* para un sistema de inventarios.
4. **Unidad 4, actividad 4. Adjuntar archivo.** Investiga el tipo de enfoque que utiliza una empresa determinada para desarrollar sus sistemas de información.
5. **Unidad 4, actividad 5.** Selecciona un problema sencillo que pueda ser solucionado a través de la implementación de un algoritmo. Desarróllalo y convierte el algoritmo resultante en un [programa de computadora](#) escrito en lenguaje C.

ACTIVIDAD INTEGRADORA

LO QUE APRENDI



Adjuntar archivo

Retoma la actividad del apartado “Lo que sé” y complementa tu concepto de programación con lo estudiado en esta unidad y la información de otras fuentes (en este caso, no olvides citarlas).

CUESTIONARIO DE REFORZAMIENTO



Adjuntar archivo. Responde las siguientes preguntas.

1. ¿Qué entiendes por una prueba de escritorio?
2. ¿Qué es un compilador?
3. ¿Qué es un diccionario de datos?
4. Explica la expresión “el programa como la expresión computable del algoritmo”.
5. ¿Cuáles son las estructuras de control básicas?
6. ¿Qué establece el teorema de la estructura?
7. Enuncia cinco ventajas de la programación estructurada.
8. ¿Qué es la modularidad?
9. ¿Qué es el refinamiento progresivo?
10. ¿Qué es el procesamiento regresivo?

EXAMEN PARCIAL

(de autoevaluación)



I. Relaciona ambas columnas.

<p>___ 1. Cargar datos muestra y seguir la lógica marcada por el diagrama o el pseudocódigo.</p> <p>___ 2. Se diagraman los pasos del problema.</p> <p>___ 3. Verifica la sintaxis del código fuente en busca de errores.</p> <p>___ 4. Se ingresan datos muestra para el análisis de los resultados.</p> <p>___ 5. Traduce cada gráfico del diagrama o línea del pseudocódigo a una instrucción de algún lenguaje de programa.</p>	<p>a) Compilación</p> <p>b) Codificación</p> <p>c) Prueba de escritorio</p> <p>d) Prueba del programa</p> <p>e) Diseño del algoritmo</p>
---	--

II. Relaciona ambas columnas.

<p>___ 1. Estructuras de control básicas.</p> <p>___ 2. Ventajas de la programación estructurada.</p> <p>___ 3. Es una característica importante en una programa estructurado.</p> <p>___ 4. La programación estructurada está basada en:</p> <p>___ 5. Serie de instrucciones que se ejecutan sucesivamente</p>	<p>a) Productividad del programador</p> <p>b) Teorema de la estructura</p> <p>c) Iteración, selección, secuencia</p> <p>d) Ser leído en secuencia</p> <p>e) Secuencia</p>
--	---

III. Responde verdadero (V) o falso (F).

	V	F
1. Los procedimientos devuelven un resultado.	<input type="radio"/>	<input type="radio"/>
2. Las funciones devuelven una tarea específica.	<input type="radio"/>	<input type="radio"/>
3. Los procedimientos son los más usados en los lenguajes de programación.	<input type="radio"/>	<input type="radio"/>
4. Las funciones son las más usadas en los lenguajes de programación.	<input type="radio"/>	<input type="radio"/>
5. El refinamiento progresivo se enfoca a la asociación e identificación de características comunes entre los casos para diseñar un modelo que represente la solución para todos ellos.	<input type="radio"/>	<input type="radio"/>
6. La técnica <i>top-down</i> o diseño descendente consiste en establecer una serie de niveles de mayor a menor complejidad.	<input type="radio"/>	<input type="radio"/>
7. El procesamiento regresivo es una técnica de análisis y diseño de algoritmos basado en la división del problema principal en problemas más simples.	<input type="radio"/>	<input type="radio"/>
8. La ventaja del diseño ascendente es que su desarrollo es mucho más económico que el descendente.	<input type="radio"/>	<input type="radio"/>
9. El diseño ascendente identifica subalgoritmos que necesitan computarizarse conforme vayan apareciendo, su análisis y su codificación, para satisfacer el problema inmediato.	<input type="radio"/>	<input type="radio"/>

IV. Responde verdadero (V) o falso (F).

	V	F
1. La compilación es un programa para convertir un código fuente a un programa ejecutable.	<input type="radio"/>	<input type="radio"/>
2. Si en las pruebas del programa se detectan errores, sólo se tienen que volver a compilar el programa.	<input type="radio"/>	<input type="radio"/>
3. Las estructuras <i>MIENTRAS</i> y <i>HASTA QUÉ</i> son condicionales iterativas.	<input type="radio"/>	<input type="radio"/>
4. El teorema de la estructura sólo hace referencia a las estructuras de control de secuencia, selección e iteración.	<input type="radio"/>	<input type="radio"/>
5. Un programa estructurado contiene instrucciones de desvío del flujo de control.	<input type="radio"/>	<input type="radio"/>
6. Sólo hay un tipo de módulo, y es la función.	<input type="radio"/>	<input type="radio"/>
7. Un procedimiento devuelve una tarea; y una función, un resultado.	<input type="radio"/>	<input type="radio"/>
8. El refinamiento progresivo contiene al procedimiento más costoso, pero el más eficiente para integrar los módulos de un sistema.	<input type="radio"/>	<input type="radio"/>
9. El <i>bottom-up</i> tiene la limitación de duplicar esfuerzos al introducir datos, ya que se incorporan al sistema muchos datos carentes de valor.	<input type="radio"/>	<input type="radio"/>

RESPUESTAS

EXAMEN DE AUTOEVALUACIÓN



En este apartado encontrarás las respuestas al examen por unidad.

Unidad 4	
I. Solución	
1.	c
2.	e
3.	a
4.	d
5.	b

Unidad 4	
II. Solución	
1.	c
2.	a
3.	d
4.	b
5.	e

Unidad 4	
III. Solución	
1.	F
2.	F
3.	F
4.	V
5.	F
6.	V
7.	F
8.	V
9.	V

Unidad 4	
IV. Solución	
1.	F
2.	F
3.	V
4.	V
5.	F
6.	F
7.	V
8.	V
9.	V

Evaluación de algoritmos

OBJETIVO PARTICULAR

Al finalizar la unidad, el alumno podrá identificar el algoritmo que solucione más eficientemente al problema en cuestión, documentarlo en futuras revisiones y llevar a efecto el mantenimiento preventivo, correctivo y adaptativo para su óptima operación.

TEMARIO DETALLADO

(16 HORAS)

5.1. Refinamiento progresivo

5.2 Depuración y prueba

5.3 Documentación del programa

5.4 Mantenimiento de programas

ACTIVIDAD DIAGNÓSTICA

LO QUE SÉ



Adjuntar archivo

De acuerdo a lo que aprendiste en las unidades anteriores, elabora un cuadro sinóptico con las principales características de un algoritmo; enfatiza los conceptos de la evaluación de un algoritmo.

ACTIVIDADES DE APRENDIZAJE



Unidad 5, actividad inicial. *Adjuntar archivo.* A partir del estudio de la bibliografía específica sugerida, elabora un mapa conceptual u [organizador gráfico](#) con los temas de la unidad. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

1. **Unidad 5, actividad 1. *Adjuntar archivo.*** Elabora un cuadro comparativo de evaluación de métodos de ordenación y determinar su eficiencia con base en la complejidad de sus algoritmos y la calidad de datos.
2. **Unidad 5, actividad 2. *Adjuntar archivo.*** Elabora un pequeño manual con los pasos para documentar programas de tal forma que te sirva como un estándar.
3. **Unidad 5, actividad 3. *Adjuntar archivo.*** Investiga en una empresa que conozcas el procedimiento que emplean para llevar a cabo el mantenimiento de sus programas o sistemas de información, y realiza un reporte.

ACTIVIDAD INTEGRADORA

LO QUE APRENDÍ



Adjuntar archivo.

A partir de la revisión general de esta unidad y la bibliografía específica sugerida, elabora un mapa mental u [organizador gráfico](#) que te facilite el estudio y comprensión de los temas analizados. Puedes auxiliarte de algunos programas como Mindjet [MindManager](#).

CUESTIONARIO DE REFORZAMIENTO



Adjuntar archivo. Responde las siguientes preguntas.

1. ¿Qué es la evaluación de algoritmos?
2. Según la forma como se encuentran los datos de entrada a un algoritmo, ¿qué significan el caso peor, el mejor y el promedio?
3. ¿Qué se entiende por depuración de programas?
4. ¿Cuáles son los errores lógicos y los de sintaxis?
5. ¿Qué es la prueba de programas?
6. Enlista cinco elementos que se verifican en la prueba de programas.
7. Enumera cinco elementos que se deben incluir en la documentación de un programa.
8. ¿Para qué sirve el mantenimiento de programas?
9. ¿Qué es el mantenimiento preventivo?
10. ¿Qué es el mantenimiento correctivo?

EXAMEN PARCIAL

(Autoevaluación)



I. Relaciona ambas columnas.

<p>___ 1. Es el proceso de identificación y corrección de errores de programación.</p>	<p>a) Sintaxis b) Depuración c) Prueba de programas d) Cálculo</p>
<p>___ 2. Tiene como propósito asegurar que el programa produce los resultados definidos en las especificaciones funcionales.</p>	
<p>___ 3. Los errores más sencillos de detectar son de...</p>	
<p>___ 4. Los errores más difíciles de detectar son de...</p>	

II. Relaciona ambas columnas.

<p>___ 1. Es una extensión de la documentación del sistema.</p>	<p>a) Lista de programas b) Documentación de programas c) Especificación de programas</p>
<p>___ 2. El programador debe trabajar conjuntamente con ella.</p>	
<p>___ 3. Incluye copia de la última compilación del programa con todas las opciones.</p>	

III. Relaciona ambas columnas.

<p>_____ 1. Los programas no tienen error, pero se requiere alguna actualización por una nueva versión del programa. Esto se refiere al mantenimiento de tipo...</p> <p>_____ 2. La revisión, la codificación para depurar y compilar un programa se realiza en el mantenimiento...</p> <p>_____ 3. La regeneración de índices y registros, realizar respaldos, verificar la integridad de los programas, actualizar porcentajes y tablas de datos, son actividades del mantenimiento...</p>	<p>a) Preventivo</p> <p>b) Adaptativo</p> <p>c) Correctivo</p>
--	--

IV. Responde verdadero (V) o falso (F).

	V	F
1. El algoritmo de selección va intercambiando elementos del arreglo hasta que esté ordenado.	<input type="radio"/>	<input type="radio"/>
2. El algoritmo de inserción va seleccionado los elementos menores al actual y los intercambia.	<input type="radio"/>	<input type="radio"/>
3. El propósito de la evaluación de un algoritmo es medir su desempeño.	<input type="radio"/>	<input type="radio"/>
4. En la evaluación del algoritmo, solamente debe considerarse el tiempo de proceso.	<input type="radio"/>	<input type="radio"/>
5. Para medir la complejidad de un algoritmo, no es necesario utilizar funciones matemáticas.	<input type="radio"/>	<input type="radio"/>
6. El termino <i>debugging</i> significa 'eliminación de bichos'.	<input type="radio"/>	<input type="radio"/>
7. Un error lógico es cuando un programa tiene errores de sintaxis.	<input type="radio"/>	<input type="radio"/>

8. El compilador es un programa que facilita la detección y corrección de errores.
9. Para realizar pruebas al programa, utilizará cualquier tipo de datos, tanto correctos como incorrectos.
10. En las pruebas al programa, se verificarán todos los cálculos que realice el programa.
11. El objetivo de la documentación de programas es familiarizar al usuario final con lo que hacen los programas.
12. El programador puede aplicar su criterio para cualquier cambio que se presente en las especificaciones del programa.
13. No es necesario incluir el diccionario de datos en la documentación de programas.

RESPUESTAS

(Examen de Autoevaluación)



En este apartado encontrarás las respuestas al examen por unidad.

Unidad 5
I. Solución
1. b
2. c
3. a
4. d

Unidad 5
II. Solución
1. b
2. c
3. a

Unidad 5
III. Solución
1. b
2. c
3. a

Unidad 5
IV. Solución
1. F
2. F
3. V
4. F
5. F
6. V
7. F
8. V
9. V
10. V
11. F
12. F
13. F



**Facultad de Contaduría y Administración
Sistema Universidad Abierta y Educación a Distancia**