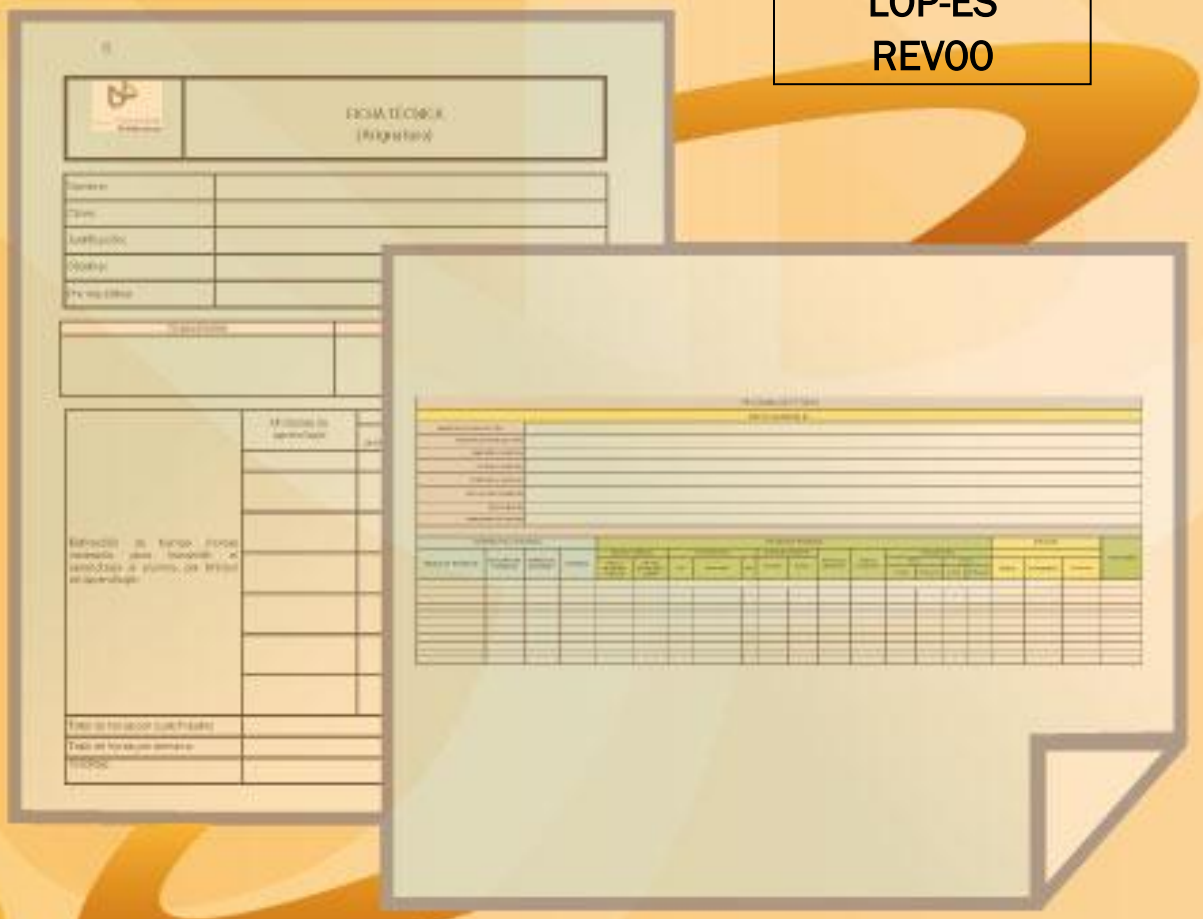
Subsistema de Universidades Politécnicas

Manual de Asignatura

LOP-ES
REV00



INGENIERÍA INDUSTRIAL

LÓGICA DE PROGRAMACIÓN

DIRECTORIO

Secretario de Educación Pública

Mtro. Alonso Lujambio Irazábal

Subsecretario de Educación Superior

Dr. Rodolfo Tuirán Gutiérrez

Coordinadora de Universidades Politécnicas

Mtra. Sayonara Vargas Rodríguez

ORIGINAL

PÁGINA LEGAL

Participantes

M.I. Rodolfo Rafael Medina Ramírez - Universidad Politécnica de Aguascalientes

ORIGINAL

Primera Edición: 2011

DR © 2011 Coordinación de Universidades Politécnicas.

Número de registro:

México, D.F.

ISBN----------

ÍNDICE

INTRODUCCIÓN 1

PROGRAMA DE ESTUDIOS 2

FICHA TÉCNICA 3

DESARROLLO DE LA PRÁCTICA O PROYECTO 5

INSTRUMENTOS DE EVALUACIÓN 9

GLOSARIO 22

BIBLIOGRAFÍA 23

ORIGINAL

INTRODUCCIÓN

El presente manual es una guía para la asignatura de Lógica de Programación, materia clave en la formación de un Ingeniero Industrial ya que se centra en el desarrollo del pensamiento analítico y razonamiento lógico-matemático requeridos para analizar problemas y proponer soluciones a partir de la aplicación del método científico.

Su objetivo es el de diseñar soluciones a partir del análisis de problemas y el diseño de una solución lógica al problema a través de un algoritmo, sentando las bases para su posterior implementación por medio de una computadora.

Esta materia es fundamental para aquellas asignaturas que hacen uso de la programación de computadoras, como Métodos Numéricos, Investigación de Operaciones, Simulación y todas aquellas disciplinas que aprovechen la programación para automatizar procedimientos de solución.

La programación de una computadora es una competencia genérica de toda ingeniería y la Ingeniería Industrial incorpora este recurso a su acervo tecnológico intelectual, tanto para desarrollar su razonamiento abstracto, como para producir respuestas eficaces en distintos ámbitos de la producción de bienes y servicios.

ORIGINAL

PROGRAMA DE ESTUDIOS

PROGRAMA DE ESTUDIO	
DATOS GENERALES	
NOMBRE DEL PROGRAMA EDUCATIVO:	Ingeniería Industrial
OBJETIVO DEL PROGRAMA EDUCATIVO:	Formar profesionistas capaces de planear, diseñar, instalar, operar, analizar y mejorar procesos productivos integrados por factor humano, materiales, información, tecnología, energía y recursos financieros, a través de la conducción de procesos de cambio y de mejora continua con una perspectiva integradora y estratégica; con actitud creativa, emprendedora y respetuosa del individuo y el medio ambiente, ajustando su desempeño a los cambios que requiere la sociedad.
NOMBRE DE LA ASIGNATURA:	Lógica de Programación
CLAVE DE LA ASIGNATURA:	LOP-ES
OBJETIVO DE LA ASIGNATURA:	El alumno será capaz de resolver problemas específicos diseñando el algoritmo de solución y codificándolo en un lenguaje de programación estructurado, analizando el comportamiento en una computadora.
TOTAL HRS. DEL CUATRIMESTRE:	75
FECHA DE EMISIÓN:	Julio, 2011
UNIVERSIDADES PARTICIPANTES:	Universidad Politécnica de Aguascalientes, Universidad Politécnica de Altamira, Universidad Politécnica de la Región Ribereña, Universidad Politécnica de San Luis Potosí, Universidad Politécnica del Estado de Morelos, Universidad Politécnica de Tlaxcala, Universidad Politécnica del Valle de México, Universidad Politécnica del Valle de Toluca y Universidad Politécnica de Zacatecas

CONTENIDOS PARA LA FORMACIÓN			ESTRATEGIA DE APRENDIZAJE													EVALUACIÓN		OBSERVACIÓN
			TÉCNICAS SUGERIDAS		ESPACIO EDUCATIVO			MOVILIDAD FORMATIVA		MATERIALES REQUERIDOS	EQUIPOS REQUERIDOS	TOTAL DE HORAS						
												TEÓRICA		PRÁCTICA				
UNIDADES DE APRENDIZAJE	RESULTADOS DE APRENDIZAJE	EVIDENCIAS	PARA LA ENSEÑANZA (PROFESOR)	PARA EL APRENDIZAJE (ALUMNO)	AULA	LABORATORIO	OTRO	PROYECTO	PRÁCTICA			Presencial	NO Presencial	Presencial	NO Presencial	TÉCNICA	INSTRUMENTO	
I. Conceptos básicos de algoritmos	Al completar la unidad de aprendizaje el alumno será capaz de: * Identificar los conceptos básicos y metodología para la solución de problemas. * Resolver problemas mediante diagramas de flujos.	**EP1**: Resuelve un problema utilizando un diagrama de flujo. **EC1**. Resuelve cuestionario referente a la secuencia de un diagrama de flujo en particular y su resultado.	Discusión dirigida Lluvia de ideas	Investigación documental Elaboración de mapas mentales Exposición	x	x	N/A	N/A	N/A	Material bibliográfico e impreso.	Computadora, proyector, pizarrón interactivo, pizarrón	5	0	5	0	Documental	* Lista de cotejo para diagrama de flujo. * Cuestionario de diagrama de flujo.	
II. Programación básica	Al completar la unidad de aprendizaje el alumno será capaz de: * Identificar las funciones básicas y sintáxis de un lenguaje de programación. * Codificar algoritmos en un lenguaje de programación.	**ED1**: Exposición sobre las funciones básicas y sintáxis de lenguaje de programación **EP1:** Realiza un programa utilizando un lenguaje de programación para resolver un problema específico.	Exposición Ejemplos ilustrativos	Investigación documental Práctica	x	x	N/A	N/A	N/A			8	0	8	3	Documental Campo	* Guía de observación para exposición. * Lista de cotejo para programa computacional.	
III. Estructuras algorítmicas de control y datos	Al completar la unidad de aprendizaje el alumno será capaz de: * Crear programas que manejen sentencias de control. * Crear programas que apliquen arreglos y matrices.	**EP1**: Realiza un programa utilizando sentencias de Control. **EP2**: Realiza un programa utilizando Arreglos y Matrices	Aprendizaje basado en problemas	Investigación documental Práctica	x	x	N/A	N/A	N/A			8	0	8	4	Documental	* Lista de cotejo para programa computacional. (EP1) * Lista de cotejo para programa computacional. (EP2)	
IV Funciones	Al completar la unidad de aprendizaje el alumno será capaz de: * Diseñar componentes reutilizables de soluciones a través de funciones.	**EP1**. Realiza un reporte de práctica en la que resuelva la problemática planteada, elaborando programas modulares mediante funciones.	Exposición, ejemplos ilustrativos, aprendizaje basado en problemas.	Aprendizaje basado en problemas, dinámica de grupos, uso de diagramas, ilustraciones y esquemas	X	X	N/A	N/A	Práctica de Funciones			4	0	4	3	Documental	* Lista de cotejo para reporte de práctica de funciones.	
V Apuntadores	Al completar la unidad de aprendizaje el alumno será capaz de: * Elaborar programas en los que hagan uso de apuntadores para el direccionamiento físico.	**EC1**. Resuelve un cuestionario sobre estructuras de datos. **EP1.** Realiza reporte de práctica de la implementación de estructuras de datos dinámicas, mediante apuntadores.	Instrucción programada, discusión dirigida, aprendizaje basado en problemas, técnica de la pregunta, instrucción programada.	Aprendizaje basado en problemas, dinámica de grupos, exposición por parte del alumno.	X	X	N/A	N/A	Práctica de Apuntadores			5	0	5	5	Documental	* Cuestionario guía para apuntadores. * Lista de cotejo para reporte de práctica de apuntadores.	

ORIGINAL

Subsistema de Universidades Politécnicas	**FICHA TÉCNICA** LÓGICA DE PROGRAMACIÓN

Nombre:	LÓGICA DE PROGRAMACIÓN
Clave:	FUP-ES
Justificación:	La asignatura permite al alumno plantear metodológicamente la solución a problemas susceptibles de ser codificados en algún lenguaje de programación a través del manejo de técnicas estructuradas de diseño y formulación de algoritmos y generar un programa de computadora para evaluar su comportamiento mediante la realización de pruebas.
Objetivo:	El alumno será capaz de resolver problemas específicos diseñando el algoritmo de solución y codificándolo en un lenguaje de programación estructurado, analizando el comportamiento en una computadora.
Habilidades:	Capacidad de análisis Pensamiento y enfoque sistémico. Interpretación de datos técnicos. Uso de tecnologías. Saber trabajar en equipo.
Competencias genéricas a desarrollar:	Capacidades para análisis y síntesis; para aprender; para resolver problemas; para aplicar los conocimientos en la práctica; para adaptarse a nuevas situaciones; para cuidar la calidad; para gestionar la información; y para trabajar en forma autónoma y en equipo.

Capacidades a desarrollar en la asignatura	Competencias a las que contribuye la asignatura
Ejecutar las corridas de simulación mediante programas de cómputo para identificar áreas de mejora	Simular el modelo de la situación a mejorar aplicando los principios de simulación y programas de cómputo para identificar áreas de mejora.

ORIGINAL

	Unidades de aprendizaje	HORAS TEORÍA		HORAS PRÁCTICA	
		Presencial	No presencial	Presencial	No Presencial
Estimación de tiempo (horas) necesario para transmitir el aprendizaje al alumno, por Unidad de Aprendizaje:	Conceptos básicos de algoritmos	5	0	5	0
	Programación básica	8	0	8	3
	Estructuras algorítmicas de control y datos	8	0	8	4
	Funciones	4	0	4	3
	Apuntadores	5	0	5	5
		30	0	30	15
Total de horas por cuatrimestre:	75				
Total de horas por semana:	5				
Créditos:	5				

ORIGINAL

Subsistema de Universidades Politécnicas	**DESARROLLO DE LA PRÁCTICA O PROYECTO**

Nombre de la asignatura:	Lógica de Programación		
Nombre de la Unidad de Aprendizaje:	Funciones		
Nombre de la práctica o proyecto:	Práctica de funciones		
Número:	1	Duración (horas) :	5
Resultado de aprendizaje:	• Diseñar componentes reutilizables de soluciones a través de funciones.		
Requerimientos (Material o equipo):	Software para desarrollar diagramas de flujo, compilador de lenguaje de programación, editor de programas de cómputo, computadora.		

Actividades a desarrollar en la práctica:

<u>El profesor</u> solicita a los estudiantes que resuelvan los problemas listados después de las instrucciones para los estudiantes y brinda atención para resolver dudas y proporciona orientación técnica para la formulación de las soluciones.

Instrucciones: El estudiante desarrolla el diagrama de flujo y el código de programación para los siguientes problemas.

1) Una función que calcule el cuadrado de un número.
2) Una función que calcule la raíz cuadrada de un número.
3) Una función que calcule cualquier potencia de cualquier número.
4) Una función que calcule el determinante de una matriz.
5) Una función que calcule la suma de dos matrices.
6) Una función que calcule el producto de dos matrices.
7) Una función que reciba una cantidad en pesos y regrese su equivalente en letras, por ejemplo, que reciba como entrada $ 30.50 y regrese treinta pesos 50/100 M.N.
8) Una función que convierta números arábigos a romanos.
9) Una función para convertir números entre los sistemas numéricos decimal, binario, octal y hexadecimal.
10) Una función que busque un elemento en una lista y en caso de encontrarlo, que regrese un apuntador a la posición del elemento en la lista.

<u>El profesor</u> Revisa las soluciones y presenta al grupo las mejores que hayan producido los estudiantes.

ORIGINAL

Evidencias a las que contribuye el desarrollo de la práctica: **EP1.** Realiza un reporte de práctica en la que resuelva la problemática planteada, elaborando programas modulares mediante funciones.

ORIGINAL

Subsistema de Universidades Politécnicas	**DESARROLLO DE LA PRÁCTICA O PROYECTO**

Nombre de la asignatura:	Lógica de Programación		
Nombre de la Unidad de Aprendizaje:	Apuntadores		
Nombre de la práctica o proyecto:	Práctica de Apuntadores		
Número:	2	Duración (horas) :	7
Resultado de aprendizaje:	• Elaborar programas en los que hagan uso de apuntadores para el direccionamiento físico.		
Requerimientos (Material o equipo):	Software para desarrollar diagramas de flujo, compilador de lenguaje de programación, editor de programas de cómputo, computadora.		

Actividades a desarrollar en la práctica:

<u>El profesor</u> solicita a los estudiantes que resuelvan los problemas listados después de las instrucciones para los estudiantes y brinda atención para resolver dudas y proporciona orientación técnica para la formulación de las soluciones.

Instrucciones: El estudiante desarrolla el diagrama de flujo y el código de programación para los siguientes problemas.

1. Desarrolla un programa que permita dar altas, bajas y cambios de clientes de un negocio y que permita ver la lista ordenada de clientes con distintos criterios: apellido paterno; fecha de nacimiento; colonia.
2. Desarrolla un programa que permita crear y mantener un catálogo de artículos de una empresa. El programa permitirá consultar el catálogo con base en: el nombre del artículo; las existencias; nivel de reorden; categoría del artículo.
3. Desarrolla un programa que use los catálogos desarrollados en los ejercicios uno y dos. Este programa registrará las ventas de los artículos. El programa actualizará las existencias de los

ORIGINAL

productos a partir de las ventas. El programa permitirá consultar las ventas con base en: el cliente; el artículo; el día de venta; el monto mínimo de venta. **El profesor** Revisa las soluciones y presenta al grupo las mejores que hayan producido los estudiantes.
Evidencias a las que contribuye el desarrollo de la práctica: **EP1.** Realiza reporte de práctica de la implementación de estructuras de datos dinámicas, mediante apuntadores.

ORIGINAL

Instrumentos de Evaluación

ORIGINAL

Subsistema de Universidades Politécnicas	**LISTA DE COTEJO PARA DIAGRAMAS DE FLUJO** **U1, EP1**

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO: UNIDAD 1, EP1		FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		
INSTRUCCIONES		
Revise las actividades que se solicitan y marque en los apartados "SI" cuando la evidencia se cumple; en caso contrario marque "NO". En la columna "OBSERVACIONES" indicaciones que puedan ayudar al alumno a saber cuáles son las condiciones no cumplidas, si fuese necesario.		

Valor del reactivo	Característica a cumplir del (reactivo)	*CUMPLE*	
		SI	*NO*
5%	Presentación. Buena presentación orden y limpieza.		
5%	Presentación. Portada. (Nombre de la escuela o logotipo, Carrera, Asignatura, Nombre del Docente, Nombre (s) de alumno (s), Grupo, Lugar y Fecha de entrega).		
30%	Método de solución. La metodología para resolver el problema tiene una secuencia lógica, resuelve una familia de problemas adecuada para los propósitos del análisis y responde a las situaciones que no cubre con un mensaje adecuado.		
5%	Diagrama de flujo. Se piden los datos de entrada necesarios y se reportan los resultados esperados.		
10%	Diagrama de flujo. Hay un inicio. Todos los flujos conducen al final del algoritmo.		

ORIGINAL

20%	Diagrama de flujo. Se respetan las reglas de uso de los símbolos que integran el diagrama de flujo y se usan de manera pertinente.		
25%	Ejemplos de pruebas efectuadas a la corrida. Muestran la funcionalidad crítica del diagrama de flujo.		
100%	Calificación		

ORIGINAL

Subsistema de Universidades Politécnicas	CUESTIONARIO DE DIAGRAMAS DE FLUJO U1, EC1

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO:	NOMBRE DEL PROYECTO:	FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		
INSTRUCCIONES		
Contesta lo que se te pide.		

REACTIVO		*CUMPLE*	
		V	*F*
____	Un algoritmo es una serie de pasos organizados que describe el proceso que debe seguir, para dar solución a un problema específico.		
--------	El algoritmo no debe tener un punto particular de inicio.		
____	Una de las características del algoritmo es que debe ser definido, es decir no debe permitir dobles interpretaciones.		
____	Una de las características del algoritmo es que debe ser finito, es decir claro, fácil de interpretar y de entender.		
____	Los algoritmos cualitativos son aquellos en los que se utilizan cálculos numéricos para definir los pasos del proceso.		
____	El lenguaje algorítmico es una serie de símbolos y reglas que se utilizan para describir de manera explícita un proceso.		
____	La única técnica para la formulación de algoritmos es el diagrama de flujo de datos		
____	*El diagrama de flujo es la representación gráfica de un algoritmo.*		

ORIGINAL

____	*Prueba de Escritorio:* Consiste en dar diferentes datos de entrada al diagrama y seguir la lógica del programa hasta tener un resultado verídico o si es necesario hacer ajustes al programa y volver a realizar la prueba.		
CALIFICACION:			

ORIGINAL

GUÍA DE OBSERVACIÓN PARA LA EVALUACION DE EXPOSICION POR EQUIPO

Lógica de Programación

NUMERO DE EQUIPO ____ GRUPO ____ FECHA ____

TEMA: Funciones básicas y sintaxis de un lenguaje de programación

EVALUACION INDIVIDUAL DEL ALUMNO

| Nombre del integrante | | |

Subsistema de Universidades Politécnicas	LISTA DE COTEJO PARA PROGRAMA COMPUTACIONAL U2, EP1 / U3, EP1 Y EP2

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO: **UNIDAD 2, EP1; UNIDAD 3, EP1 y EP2.**		FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		
INSTRUCCIONES		
Revise las actividades que se solicitan y marque en los apartados "SI" cuando la evidencia se cumple; en caso contrario marque "NO". En la columna "OBSERVACIONES" indicaciones que puedan ayudar al alumno a saber cuáles son las condiciones no cumplidas, si fuese necesario.		

Valor del reactivo	Característica a cumplir del (reactivo)	*CUMPLE*	
		SI	*NO*
5%	Presentación. Buena presentación orden y limpieza.		
5%	Presentación. Portada. (Nombre de la escuela o logotipo, Carrera, Asignatura, Nombre del Docente, Nombre (s) de alumno (s), Grupo, Lugar y Fecha de entrega).		
20%	Diagrama de flujo. Se entrega un diagrama de flujo correcto y validado con sus corridas de escritorio.		
5%	Código de programación. Los datos de entrada necesarios se solicitan y validan adecuadamente. Se reportan todos los resultados esperados con un formato adecuado.		
10%	Código de programación. El programa siempre termina de manera normal de acuerdo con la petición de proceso del usuario, no se cicla.		

ORIGINAL

30%	Código de programación. La metodología para resolver el problema tiene una secuencia lógica, resuelve una familia de problemas adecuada para los propósitos del análisis y responde a las situaciones que no cubre con un mensaje adecuado.		
25%	Ejemplos de pruebas efectuadas a la corrida. Muestran la funcionalidad crítica del programa.		
100%	Calificación		

ORIGINAL

Subsistema de Universidades Politécnicas	**LISTA DE COTEJO PARA REPORTE DE PRÁCTICA DE FUNCIONES** **U4, EP1**

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO: **UNIDAD 4, EP1.**		FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		

INSTRUCCIONES

Revise las actividades que se solicitan y marque en los apartados "SI" cuando la evidencia se cumple; en caso contrario marque "NO". En la columna "OBSERVACIONES" indicaciones que puedan ayudar al alumno a saber cuáles son las condiciones no cumplidas, si fuese necesario.

Valor del reactivo	**Característica a cumplir del (reactivo)**	***CUMPLE***	
		SI	***NO***
5%	Presentación. Buena presentación orden y limpieza.		
5%	Presentación. Portada. (Nombre de la escuela o logotipo, Carrera, Asignatura, Nombre del Docente, Nombre (s) de alumno (s), Grupo, Lugar y Fecha de entrega).		
5%	Ejercicios completos. Se entregaron todos los ejercicios solicitados.		
20%	Diagrama de flujo. Cada ejercicio muestra el diagrama de flujo de cada función solicitada.		
30%	Código de programación. Cada función está codificada de manera que responde a las especificaciones de diseño de manera adecuada. La función verifica que los datos de entrada sean adecuados y regresa un código de error en caso de que la respuesta no se pueda generar.		

ORIGINAL

10%	Código de programación. La función incluye la documentación sobre los parámetros de entrada esperados, la salida producida y los posibles códigos de error a generar.		
25%	Ejemplos de pruebas efectuadas a la corrida. Muestran la funcionalidad crítica de la función.		
100%	Calificación		

ORIGINAL

Subsistema de Universidades Politécnicas	CUESTIONARIO GUÍA PARA APUNTADORES U5, EC1

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO:	NOMBRE DEL PROYECTO:	FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		
INSTRUCCIONES		
Contesta lo que se te pide.		

REACTIVO	*CUMPLE*	
	V	*F*
____ Un apuntador es un recurso estático para manejar memoria de cómputo.		
-------- La liberación de memoria de un apuntador dinámico de memoria es una función automática que hace el compilador para el programador.		
____ Un apuntador dinámico de memoria permite superar las limitantes de tamaño de memoria que imponen los arreglos estáticos.		
____ Una lista dinámica puede ser implementada a través de apuntadores estáticos.		
____ Actualizar los apuntadores a los elementos previo y siguiente en un procedimiento de borrado de una lista dinámica es responsabilidad del programador.		
____ Los lenguajes de programación evolucionaron al ofrecer apuntadores dinámicos como parte de sus recursos.		
____ Con apuntadores dinámicos es posible mantener de manera permanente diversos ordenamientos de listas sin multiplicar las instancias de los componentes de las listas.		
CALIFICACION:		

ORIGINAL

Subsistema de Universidades Politécnicas	LISTA DE COTEJO PARA REPORTE DE PRÁCTICA DE APUNTADORES U5, EP1

UNIVERSIDAD POLITÉCNICA _____		
DATOS GENERALES DEL PROCESO DE EVALUACIÓN		
NOMBRE DEL ALUMNO:	MATRICULA:	FIRMA DEL ALUMNO:
PRODUCTO: **UNIDAD 5, EP1.**		FECHA:
ASIGNATURA: LÓGICA DE PROGRAMACIÓN		PERIODO CUATRIMESTRAL:
NOMBRE DEL DOCENTE:		
INSTRUCCIONES		
Revise las actividades que se solicitan y marque en los apartados "SI" cuando la evidencia se cumple; en caso contrario marque "NO". En la columna "OBSERVACIONES" indicaciones que puedan ayudar al alumno a saber cuáles son las condiciones no cumplidas, si fuese necesario.		

Valor del reactivo	Característica a cumplir del (reactivo)	*CUMPLE*	
		SI	*NO*
5%	Presentación. Buena presentación orden y limpieza.		
5%	Presentación. Portada. (Nombre de la escuela o logotipo, Carrera, Asignatura, Nombre del Docente, Nombre (s) de alumno (s), Grupo, Lugar y Fecha de entrega).		
5%	Ejercicios completos. Se entregaron todos los ejercicios solicitados en la práctica de apuntadores.		
20%	Diagrama de flujo. Cada ejercicio muestra el diagrama de flujo de las funciones que manipulan apuntadores.		
30%	Código de programación. La representación de los datos permite su manipulación con apuntadores para hacer un uso eficiente de memoria. Las funciones para manipular la estructura de datos aprovechan los apuntadores y hacen un uso eficaz y eficiente de la memoria. La función verifica que los datos de entrada sean adecuados y regresa un código de error en caso de que la respuesta no se pueda generar.		

ORIGINAL

10%	Código de programación. La función incluye la documentación sobre los parámetros de entrada esperados, la salida producida y los posibles códigos de error a generar.		
25%	Ejemplos de pruebas efectuadas a la corrida. Muestran la funcionalidad crítica de la función.		
100%	Calificación		

ORIGINAL

GLOSARIO

Algoritmo: Serie de pasos organizados que describe el proceso que se debe seguir para dar solución a un problema específico.

Estructuras de Control: Permiten modificar el flujo de ejecución de las instrucciones de un algoritmo.

DFD: Diagrama de flujo de datos: es la representación gráfica de un algoritmo.

Arreglos: Estructura de datos que almacena bajo el mismo nombre a una colección de datos del mismo tipo.

Matriz: Es un arreglo de m*n elementos en dos dimensiones donde m es el numero de fila y n es el numero de columnas.

Cadenas de Caracteres: Palabra ó conjunto de letras, números y símbolos.

Subprogramas: Son rutinas, procedimientos o conjuntos de instrucciones que realizan una labor específica.

Funciones: Grupo de instrucciones con un objetivo en particular y que se ejecuta al ser llamada desde otra función o procedimiento

Expresiones Aritméticas Combinación de operadores y operandos.

Expresiones Lógicas: Únicamente pueden tomar dos valores cierto ó falso. Cierto únicamente si ambas expresiones lógica son ciertas.

Lenguaje de programación: Lenguaje formal que una computadora puede reconocer a través del cual se expresa la metodología para solucionar un problema específico.

ORIGINAL

BIBLIOGRAFÍA

Básica

TÍTULO:	C++ COMO PROGRAMAR
AUTOR:	DEITEL HARVEY M. Y DEITEL PAUL
AÑO:	2008
EDITORIAL O REFERENCIA:	PEARSON
LUGAR Y AÑO DE LA EDICIÓN	2008
ISBN O REGISTRO:	9702-612-73X

TÍTULO:	PROGRAMACIÓN ESTRUCTURADA EN C
AUTOR:	GARCÍA - BERMEJO
AÑO:	2008
EDITORIAL O REFERENCIA:	PEARSON EDUCATION
LUGAR Y AÑO DE LA EDICIÓN	ENERO, 2008
ISBN O REGISTRO:	8483-224-232

TÍTULO:	FUNDAMENTOS DE PROGRAMACIÓN
AUTOR:	JOYANES AGUILAR LUIS
AÑO:	2008
EDITORIAL O REFERENCIA:	MC. GRAW HILL
LUGAR Y AÑO DE LA EDICIÓN	ESPAÑA, 2008
ISBN O REGISTRO:	9788-4481-61118

Complementaria

TÍTULO:	ESTRUCTURA DE DATOS: LIBRO DE PROBLEMAS
AUTOR:	JOYANES AGUILAR LUIS
AÑO:	2008
EDITORIAL O REFERENCIA:	MC. GRAW HILL
LUGAR Y AÑO DE LA EDICIÓN	ESPAÑA, 2008
ISBN O REGISTRO:	9788-4481-22980

TÍTULO:	PROGRAMACIÓN Y RESOLUCIÓN DE PROBLEMAS CON C++
AUTOR:	DALE NELL
AÑO:	2007
EDITORIAL O REFERENCIA:	MC. GRAW HILL
LUGAR Y AÑO DE LA EDICIÓN	ENERO, 2007
ISBN O REGISTRO:	9701-0611-04

ORIGINAL