	UNIVERSIDAD POPULAR DEL CESAR	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 1 de 6

IDENTIFICACIÓN

Nombre de la asignatura	Programación Estructurada		
Código de la asignatura	SS401		
Programa Académico	Ingeniería de Sistemas		
Créditos académicos	4		
Trabajo semanal del estudiante	Docencia directa: 4	Trabajo Independiente: 5	
Trabajo semestral del estudiante	144		
Pre-requisitos	ALGORITMO Y FUNDAMENTOS DE PROGRAMACIÓN		
Co-requisitos			
Departamento oferente	Ingeniería de Sistemas		
Tipo de Asignatura	Teórico:	Teórico-Práctico	Práctico:
Naturaleza de la Asignatura	Habilitable:		No Habilitable:X
	Validable:		No Validable:X
	Homologable:X		No Homologable:

PRESENTACIÓN

El curso provee al estudiante los conocimientos necesarios para desarrollar la lógica algorítmica; aplicar los conocimientos de un lenguaje estructurado en la resolución de problemas computacionales.


JUSTIFICACIÓN

Es importante que el estudiante de ingeniería de sistemas obtenga y desarrolle destrezas y habilidades en la solución de problemas mediante computadores, a partir de conceptos básicos de algoritmos, siendo esta una asignatura básica, que permite al estudiante la construcción adecuada y eficiente de programas estructurados.

Además se estudiarán los conceptos fundamentales en el área de programación, para que el estudiante pueda enfocar la fase preliminar de solución de problemas, en cualquier disciplina, utilizando un lenguaje de programación que permita al computador interpretar los algoritmos planteados.

OBJETIVO GENERAL

Desarrollar en el estudiante la lógica en la solución de problemas cotidianos y complejos a través de métodos algorítmicos y lenguajes de programación.

	UNIVERSIDAD POPULAR DEL CESAR	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 2 de 6

OBJETIVOS ESPECÍFICOS

- Impartir a los estudiantes los conceptos básicos teóricos y prácticos de los lenguajes de programación
- Orientar al estudiante en la forma de definir y analizar un problema de forma lógica y computacional
- Diseñar algoritmos que solucionen dichos problemas
- Codificar los algoritmos mediante lenguajes de programación

COMPETENCIAS GENERALES Y ESPECÍFICAS


- Capacidad de trabajo en equipo y liderazgo
- Capacidad para proponer soluciones a un problema algorítmicos
- Capacidad de análisis y síntesis
- Capacidad de planificación y organización
- Capacidad de utilizar adecuadamente un lenguaje de programación en la solución de una problemática
- Capacidad de comprender la naturaleza y limitaciones de los lenguajes de programación, y sus campos de aplicación
- Capacidad para utilizar las distintas técnicas y estrategias en la elaboración de algoritmos y su utilidad en problemas relacionados con manejo de información

METODOLOGÍA

La asignatura será dirigida por el docente, teniendo en cuenta los temas establecidos por la dirección del programa, a través de:

Docencia Directa: Esta estrategia corresponde a clases presenciales, dirigidas por el docente, el cual explicará y profundizará las ideas y conceptos principales de cada tema, fomentando la investigación, participación e interés del estudiante, mediante el diseño y dirección de proyectos, talleres, debates, mesas redondas, sustentaciones y socialización de: lecturas autorreguladas, mapas conceptuales y ensayos.

Trabajo Independiente: Esta estrategia corresponde al autoaprendizaje por parte del estudiante. Para ello, debe documentarse y preparar los diferentes temas de la asignatura con anticipación; teniendo en cuenta el contenido suministrado por el docente, utilizando las diferentes fuentes bibliográficas.

	UNIVERSIDAD POPULAR DEL CESAR	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 3 de 6

Horas de Asesoría: Esta estrategia corresponde a la asesoría que debe brindar el docente a los estudiantes, sobre las tareas asignadas y en horas estipuladas independientemente de las horas de docencia directa.

ESTRATEGIAS METODOLÓGICAS

Entre las estrategias pedagógicas utilizadas por el docente para impartir la asignatura se encuentran:

- **Talleres.** Esta estrategia metodológica fortalece el proceso de enseñanza- aprendizaje; el taller es una actividad práctica que promueve un espacio de reflexión y construcción del conocimiento; estos son previamente diseñados por los docentes con base a las competencias que el estudiante debe desarrollar en cada asignatura y publicados en espacios tales como: web sites, blogs, aula web o aula de clases. Las asignaturas de tipo teórico - práctico usan esta estrategia para promover el trabajo en equipo, consultas y profundización investigativa.
- **Mediaciones Virtuales.** El uso y apropiación de las tics se convierten en herramientas claves que son de apoyo al proceso de formación en el aula de clases, debido a que promueven en el estudiante la búsqueda permanente del conocimiento a través de herramientas como: plataformas virtuales- aula web, redes profesionales, sociales, web sites, aplicaciones en la nube, correo electrónico, foros y demás herramientas sincrónicas y asincrónicas que facilitan la interacción.
- **Visitas empresariales:** estas acercan al estudiante con aplicaciones y situaciones reales de la ingeniería de sistemas en los diferentes sectores productivos. Asociadas al conocimiento que el estudiante debe tener de su entorno, persiguiendo la construcción de pensamiento crítico y el aprendizaje significativo.
- **Proyecto de aula:** esta estrategia corresponde al desarrollo del proyecto guiado por el docente desde el inicio del semestre y donde el estudiante es el actor principal, quien debe identificar problemas del entorno y a través de aplicativos confiables contribuir a la optimización de los procesos.



CONTENIDO

UNIDAD 1. GENERALIDADES Y CONCEPTOS

- 1.1. Qué es un lenguaje de programación
 - 1.1.1 Lenguajes maquina
 - 1.1.2 Lenguajes de bajo nivel (ensamblador)
 - 1.1.3 Lenguaje de alto nivel traductores de lenguajes
- 1.2.1 Interpretes
- 1.2.2 Compiladores
- 1.3 Orígenes del C/C++

UNIDAD 2. INTRODUCCION A LA PROGRAMACION

- 2.1 Evolución de la programación
- 2.2 Programación lineal
- 2.3 Programación modular
- 2.4 Programación estructurada
- 2.5 Abstracción de datos
- 2.6 Programación orientada a objetos


UNIDAD 3. INTRODUCCION AL C/C++

- 3.1 Tipos de datos
- 3.2 Constantes
- 3.3 Variables
 - 3.3.1 Concepto de variables
 - 3.3.2 Definición de variables
- 3.4 Operadores aritméticos

- 3.5 Otros Operadores
- 3.6 Jerarquía de operadores
- 3.7 Como escribir un programa en C++
 - 3.7.1 Interpretación de la declaración #include
 - 3.7.2 Interpretación del uso del Void
 - 3.7.3 Interpretación de las declaraciones de agrupamiento {}
- 3.8 Tipos de asignaciones
- 3.9 Instrucciones de entrada y salida por teclado y pantalla

UNIDAD 4. TOMA DE DECISIONES

- 4.0 Operador ?
- 4.1 Instrucción If
- 4.2 Instrucción if-else
- 4.3 Instrucciones Else-if-else
- 4.4 If anidado

	UNIVERSIDAD POPULAR DEL CESAR	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 5 de 6

4.5 Instrucción Switch

4.6 Ejercicios

UNIDAD 5. INSTRUCCIONES REPETITIVAS

5.1. Instrucciones FOR

5.2. Instrucciones While

5.3. Instrucciones DO-WHILE

5.4. Ejercicios

UNIDAD 6. SUBROUTINAS (FUNCIONES)

6.1. Objetivos

6.2. Concepto de función

6.3. Parámetros por referencia y por valor

6.4. Instrucción return

6.5. Llamadas a las funciones

6.6. Librerías de funciones

6.7 Otras Funciones usuales del C++

6.7.1 Tratamiento de cadenas

6.8 Ejercicios

UNIDAD 7. ARREGLOS

7.1 Definición

7.2 Arreglos unidimensionales

7.3 Arreglos multidimensionales

7.4 Paso de arreglos como parámetros de funciones

7.5 Ejercicios

UNIDAD 8. ESTRUCTURAS O REGISTROS

8.1 Arreglos de registros

8.2 Paso de estructuras como parámetros de funciones

UNIDAD 9. ARCHIVOS


9.1 Introducción a los archivos

9.2 Operaciones Entrada/Salida

9.3 Tipos de archivos (Texto – Binario)

EVALUACIÓN

La calificación de la asignatura está organizada de la siguiente forma:

	UNIVERSIDAD POPULAR DEL CESAR	CODIGO: 201-300-PRO05-FOR01
		VERSIÓN: 1
PLAN DE ASIGNATURA		PÁG: 6 de 6

PARCIALES:

- **Primer parcial:** 30%(5% talleres, trabajos, 5% primera entrega del proyecto final y 20% parcial).
- **Segundo parcial:** 30%(5% talleres, trabajos y asistencia, 5% segunda entrega del proyecto final y parcial 20%.
- **Tercer parcial:** 40%(20% Parcial y 20% Trabajo final).

TRABAJOS

Asistencia a clases.
Ejercicios.
Exposiciones.
Talleres.

REFERENCIAS BIBLIOGRÁFICAS

- LUIS JOYANES AGUILAR. Fundamentos de Programación algoritmos, estructuras de datos y objetos. Editorial Mc Graw Hill. Cuarta edición. 2003.
- ROBERT SEDGEWICK. Algoritmos en C++. ADDISON WESLEY LONGMAN. 2000
- ADARME JAIMES, MARCO ANTONIO. Reflexión computacional: Un enfoque desde C++. ECOE EDICIONES. 2013
- CEBALLOS SIERRA, FRANCISCO JAVIER. C/C++ curso de programación. EDITOR GRUPO ALFAOMEGA. 2002
- NELL DALE Y CHIP WEEMS. Programación y resolución de problemas con C++. MCGRAW-HILL INTERAMERICANA. 2007
- PÁEZ MEJÍA, MANUEL JOSÉ. C y C++ de Afán. EDITORIAL UNIVERSIDAD DE ANTIOQUIA. 2004