

**VICERRECTORIA ACADÉMICA
FORMATO DE PRESENTACIÓN DE ASIGNATURAS**

FACULTAD:
PROGRAMA:

**INGENIERÍAS Y TECNOLÓGICAS
INGENIERÍA DE SISTEMAS**

I. PRESENTACION DE LA ASIGNATURA

Nombre de la asignatura	Programación I
Área curricular	
Requisito	Introducción a la ingeniería y lógica conjunto y grafos
Correquisito	
No. TOTAL DE CREDITO	4
Horas docencia directa	4
Código de la asignatura	
Horas de trabajo independiente	

II. JUSTIFICACIÓN DE LA ASIGNATURA

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 alumno la construcción adecuada y eficiente en el desarrollo de software.

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 como Pascal o C++, los cuales permiten que el computador interprete los algoritmos planteados.

III. OBJETIVOS

➤ **OBJETIVO GENERAL**

Desarrollar en el estudiante la lógica en la solución de problemas cotidianos y complejos a través de métodos algorítmicos y las herramientas computacionales.

➤ **OBJETIVOS ESPECÍFICOS**

- Impartir a los estudiante los conceptos básicos teóricos y prácticos de los computadores
- Orientar al estudiante la forma de definir y analizar un problema
- Diseñar algoritmos que solucionen dichos problemas
- Enseñar a estructurar un programa computacional
- Determinar si los algoritmos desarrollados dan las soluciones adecuadas
- Codificación de algoritmos mediante lenguajes de programación básicos.

IV. CONTENIDO DE LA ASIGNATURA

1. GENERALIDADES Y CONCEPTOS

- 1.1. Que 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++

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

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

INGENIERIA DE SISTEMAS

- 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

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
- 4.5 Instrucción Switch
- 4.6 Ejercicios

5. INSTRUCCIONES REPETITIVAS

- 5.1. Instrucciones FOR
- 5.1. Instrucciones While
- 5.1. Instrucciones DO-WHILE
- 5.1. Ejercicios

6. SUBROUTINAS (FUNCIONES)

- 6.1. Objetivos
- 6.2. Concepto de función
- 6.3. Parámetros, argumentos
- 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
 - 6.7.2 Manejo de Gráficos
- 6.8 Ejercicios

7. ARREGLOS

- 7.1 Definición
- 7.2 Arreglos unidimensionales
- 7.3 Arreglos multidimensionales
- 7.4 Ejercicios

8. REGISTROS

- 8.1 ARREGLOS DE REGISTROS

INGENIERIA DE SISTEMAS

9. ARCHIVOS

- 9.1 Introducción a los archivos
- 9.2 Operaciones Entrada/Salida
- 9.3 Tipos de archivos (Texto – Binario)

V. ESTRATEGIAS METODOLÓGICAS

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 y las franjas de usuario programadas en las salas de informática.

**** 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.

VI. SISTEMA DE EVALUACIÓN

Para las evaluaciones el docente tendrá en cuenta los trabajos teóricos y prácticos, individuales y en grupo, utilizando las siguientes estrategias:

- Seguimiento a los proyectos de aplicación.
- Sustentaciones de las tareas asignadas y proyectos desarrollados.
- Evaluaciones que permitan determinar los logros y falencias que se presenten los estudiantes

VII. BIBLIOGRAFÍA

LUIS JOYANES AGUILAR. Fundamentos de Programación algoritmos y estructuras de datos. Editorial Mc Graw Hill. Segunda edición. 1996.

PEREZ, José. Algoritmos, estructuras Básicas. Ediciones Uninorte 1993.

JOYANES, Luis. Problemas de metodología de la programación. MC Graw-Hill 1990

LONG, Larry. Introducción a las computadoras y al procesamiento de la Información, MC Graw-Hill. 1990.

CORREA, Guillermo. Desarrollo de Algoritmos Mc Graw-Hill. 1990

PRIETO, Alberto Introducción a la Informática. McGraw-Hill. 1992

TREMBLAY, Jeany y BURT, Richard. Introducción a la ciencia de los computadores. Un enfoque algoritmico. Mc Graw-Hill 1988.

SANDERS, Donald. Informática Presente y Futuro, 3a De. McGraw-Hill. México 1992.

MARY E.S. Loonis. Algoritmo y Estructura de datos. Practice - Hall. Hispanoamericana S.A., 1993.

