**SÍLABO**

1. **INFORMACIÓN GENERAL**

Curso: **Introducción a la Computación**

Código : BIC01

Prerrequisito : Ninguno

Dpto. Académico : Estudios Generales

Condición : Obligatorio

Ciclo Académico : 2025-2

Créditos : 2

Horas teóricas : 1 hora semanal

Horas prácticas / Laboratorio : 2 horas semanales

Sistema de Evaluación : F

Profesor del curso : Miguel Angel Navarro Neyra

1. **SUMILLA**

La asignatura pertenece al área de formación general y es de naturaleza teórico práctico. Tiene como propósito desarrollar la capacidad de aplicar las herramientas del Lenguaje de Programación a nivel estructural, que involucra conceptos, como Algoritmos, Datos, Números Aleatorios, Procesos Secuenciales, Estructuras de control, Estructuras Repetitivas, Arreglos, Cadenas y funciones que son comunes a cualquier Lenguaje de programación y la codificación o código fuente para desarrollar aplicaciones de índole Formativo.

1. **COMPETENCIAS**
2. Aplicar con destreza la programación de instrucciones de entrada y salida, operaciones, fórmulas, funciones matemáticas estándar, instrucciones de decisión, repetición y control.
3. Identificar la terminología básica y contexto de la programación estructurada.
4. Identificar las características distintivas de los lenguajes de programación procedimentales y funcionales.
5. Aplicar la capacidad para analizar, diseñar e implementar soluciones a problemas computacionales de baja y mediana complejidad.
6. Aplicar los procesos para automatizar en el Lenguaje C++, en modo Consola.
7. **UNIDADES DE APRENDIZAJE**

**UNIDAD 1. CONCEPTOS BÁSICOS DE LA COMPUTACIÓN (4 semanas)**

**Historia de la Computación:**

* Desarrollo de la computación a lo largo del tiempo.
* Hitos importantes en la historia de la tecnología.

**Conceptos Básicos de Hardware:**

* Representación de los datos bits, bytes.
* Codificación de datos.
* Componentes de una computadora (CPU, memoria, almacenamiento, etc.).
* Periféricos y dispositivos de entrada/salida.
* Microprocesadores.
* Arquitecturas modernas.

**Conceptos Básicos de Sistema Operativo:**

* Funciones y roles del sistema operativo.
* Componentes de un sistema operativo.
* Tipos de sistemas operativos (Tiempo Real, Propósito general; Windows, macOS, Linux, etc.).
* Tipo de sistemas operativos (Centralizados, distribuidos)
* Gestión de Procesos, Memoria, Periféricos, Archivos.

**Introducción a la Teoría de Compiladores:**

* Análisis léxico
* Análisis sintáctico
* Análisis sintáctico descendente
* Análisis sintáctico ascendente
* Tabla de tipos y de símbolo
* Generación del código.
* Lenguajes de programación
* Tipos de lenguajes de programación.

**UNIDAD 2. INTRODUCCIÓN A LA ALGORITMIA (3 semanas)**

**Concepto de algoritmo.**

* Concepto de algoritmo
* Definición de algoritmo y problema
* Análisis del enunciado de un problema
* Análisis del problema
* Datos de entrada
* Datos de salida
* Memoria y operaciones aritméticas y lógicas
* Teorema de la programación estructurada
* Diseño del algoritmo
* Registro de instrucciones

**Elementos de un programa.**

* Identificadores
* Variables y valores
* Tipos de datos primitivos
* Literales
* Operadores
  1. Aritméticos
  2. Lógicos y relacionales

**Elementos de un programa.**

* Expresiones
* Expresiones aritmeticológicas
* Jerarquía de operadores
* Conversión de tipos.
* Palabras reservadas
* Tipos de algoritmos
  1. Lenguaje natural (español, inglés, etc.)
  2. Diagramas de flujo
  3. Pseudocódigo
* Diagramas de flujo
  1. Terminal
  2. Entrada/salida
  3. Proceso
* Prueba de escritorio

**SEMANA 8: EXAMEN PARCIAL**

**UNIDAD 3. ESTRUCTURAS DE CONTROL (5 semanas)**

**Estructuras secuenciales**

* Pseudocódigo
* Ejercicios
* Instrucciones algorítmicas básicas
* Estructuras secuenciales
* Problemas

**Estructuras de Selección**

* Estructura selectiva Simple IF THEN
* Estructura Selectiva doble IF THEN ELSE

**Estructuras de Selección**

* Estructura selectiva múltiple IF THE ELSE IF
* Estructura selectiva de casos SWITCH

**Estructuras repetitivas**

* Introducción
* Estructura de repetición while.
* Números aleatorios

**Estructuras repetitivas**

* Estructura de repetición do...while.
* Estructura de repetición for.
* Problemas resueltos.

**Contadores y acumuladores**

* Operadores de incremento y decremento
* Operadores de asignación compleja
* Contadores y acumuladores.

**UNIDAD 4. ESTRUCTURAS DE DATOS ELEMENTALES (2 semanas)**

**Arreglos**

* Arreglos unidimensionales

**SEMANA 16: EXAMEN FINAL**

1. **METODOLOGÍA**

El curso se desarrolla en sesiones de teorías y laboratorio de cómputo. En las sesiones de teoría, el docente presenta los conceptos y aplicaciones. En las sesiones de Laboratorio, se desarrolla y prueba todo lo aprendido en la teoría. Al final del curso el alumno debe presentar y exponer un trabajo o proyecto integrador. En todas las sesiones se promueve la participación activa del alumno.

1. **LABORATORIOS**

**Laboratorio 1:** Diagramas de Flujo

**Laboratorio 2:** Aplicaciones con procesos secuenciales. Aplicaciones con Números Aleatorios

**Laboratorio 3:** Aplicaciones con estructuras selectivas simples, dobles, múltiples y anidadas

**Laboratorio 4:** Aplicaciones con estructuras de control simple, múltiple y repetitivas con WHILE y FOR

1. **SISTEMA DE EVALUACIÓN**
2. Sistema de Evaluación “F”:

EP: Examen Parcial (Peso 1)

EF: Examen Final (Peso 2)

PP: Promedio de Prácticas o Trabajos Calificados (Peso 1)

1. Cantidad de Prácticas o Trabajos Calificados cuatro (04).
2. Fórmula de Evaluación:

**Promedio Final = (EP + 2EF + PP) / 4**

PP: Promedio de Prácticas. Se obtiene del promedio aritmético de las tres (03) mejores notas de las prácticas o trabajos calificados.

1. **BIBLIOGRAFÍA**

**BIBLIOGRAFÍA BÁSICA**

1. JOYANES AGUILAR, Luis (2008) “Fundamentos de Programación Algoritmos, estructura de datos y objetos” 4ta Ed. Editorial McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A. U.
2. Introduction to Computing Explorations in Language, Logic, and Machines. David Evans University of Virginia, Version: August 19, 2011
3. Manual Basico de C++ 5a edición Stanley Lippman, vigésima edición, Addison Wesley