



**NOMBRE DEL CURSO: Introducción a la Programación y computación 1**

<b>CÓDIGO:</b>	<b>0770</b>	<b>CRÉDITOS:</b>	<b>4</b>
<b>ESCUELA:</b>	<b>Ciencias y Sistemas</b>	<b>AREA A LA QUE PERTENECE:</b>	<b>Desarrollo de Software</b>
<b>PRE REQUISITO:</b>	<b>33 créditos y 0103 Matemática Básica 2</b>	<b>POST REQUISITOS:</b>	<b>0771 Introducción a la Programación y Computación 2 0796 Lenguajes Formales y de Programación.</b>
<b>CATEGORIA:</b>	<b>Obligatorio</b>	<b>SEMESTRE:</b>	<b>Primer Semestre 2020</b>
<b>CATEDRATICO(A):</b>	<b>Ing. Neftalí Calderón</b>	<b>AUXILIAR:</b>	<b>Dénilson Argueta</b>
<b>EDIFICIO:</b>	<b>T - 7</b>	<b>SECCION:</b>	<b>E</b>
<b>SALON DEL CURSO:</b>	<b>101</b>	<b>SALON DEL LABORATORIO:</b>	<b>T-3 413</b>
<b>HORAS POR SEMANA DEL CURSO</b>	<b>4</b>	<b>HORAS POR SEMANA DEL LABORATORIO</b>	<b>2</b>
<b>DÍAS QUE SE IMPARTE EL CURSO</b>	<b>Martes y Jueves</b>	<b>DÍAS QUE SE IMPARTE EL LABORATORIO</b>	<b>Viernes</b>
<b>HORARIO DEL CURSO</b>	<b>7:10 - 8:50</b>	<b>HORARIO DEL LABORATORIO</b>	<b>7:10-8:50</b>

**DESCRIPCIÓN DEL CURSO:**

El curso es el acercamiento inicial del estudiante de la carrera de sistemas, a la programación mediante el uso de disciplinas y metodologías especializadas. El curso se fundamenta en el concepto de algoritmo para la resolución de problemas de programación, enfatizando el uso del paradigma orientado a objetos. Se introducen conceptos básicos de UML como guía para el diseño de sistemas orientados a objetos.

Se acerca al estudiante al conocimiento de los principales algoritmos de búsquedas y ordenamientos. Se cubre una parte importante de las estructuras de datos, los tipos de datos abstractos. Asimismo, el estudiante conocerá el lenguaje Java como el lenguaje oficial de programación del curso.

## **OBJETIVOS:**

### **General**

- Lograr que el estudiante adquiera la habilidad de programar y los conocimientos básicos de la programación utilizando el paradigma orientado a objetos.

### **Específico**

- Integrar al estudiante a la tecnología de la computación.
- Conocer las diferentes metodologías de programación.
- Organizar soluciones utilizando un lenguaje de programación.
- Adquirir la habilidad de hacer algoritmos.
- Aprender a elaborar diseños de clases preliminares en UML.
- Analizar los problemas con metodología orientada a objetos.
- Conocer el lenguaje Java como el primer lenguaje de programación para computadoras.

## **METODOLOGÍA:**

- Clases presenciales con contenido audiovisual, clases virtuales, lecturas, etc.
- Capacitación personalizada al estudiantado.
- Elaboración de Actividades.
  - o Tareas e Investigaciones.
  - o Prácticas.
  - o Proyectos.
  - o Evaluaciones.
  - o Evaluación Final.

## **REQUISITOS**

- Es obligatorio aprobar ambos proyectos de laboratorio con una nota de 61 puntos.
- El laboratorio se aprueba con 61 puntos.
- Las actividades a realizar en el laboratorio (tareas, investigaciones, prácticas, proyectos, evaluaciones cortas, examen final, etc.) son dependientes. Consiste en que cada actividad será dependiente de la anterior.
- **Para tener derecho a su nota final debe de entregar todas las actividades.**
- La forma de entrega de las actividades será vía Classroom, según la fecha y hora límite de entrega en base al enunciado de cada actividad.
- Para la calificación de las actividades se tomará en cuenta la presentación, calidad y contenido de ellas.

**EVALUACIÓN:**

La nota de promoción para aprobar el laboratorio es de 61 puntos de 100.

La ponderación para cada actividad de evaluación es la siguiente:

<b>Descripción</b>	<b>Punteo</b>
Practica 1	7.5
Practica 2	7.5
Practica Presencial	5
<b>Total de Practicas</b>	<b>20</b>
Proyecto 1	20
Proyecto 2 – Fase 1	10
Proyecto 2 – Fase 2	20
<b>Total Proyectos</b>	<b>50</b>
Tarea 1	1
Tarea 2	2
Tarea 3	2
Hoja de Trabajo 1	2
Hoja de Trabajo 2	4
Hoja de Trabajo 3	4
<b>Total Tareas</b>	<b>15</b>
Corto 1	2.5
Corto 2	2.5
<b>Total de Cortos</b>	<b>5</b>
<b>Examen Final</b>	<b>10</b>
<b>Total</b>	<b>100</b>

**Para ganar el laboratorio se debe de tener 80% de asistencia.**

## **CONTENIDO DE LABORATORIO**

### **1. Fundamentos de Programación 03/02/20 – 07/02/20**

- 1.1 Introducción a Algoritmos.
- 1.2 Diagramas de Flujo.
- 1.3 Introducción a la programación.

### **2. Introducción a Java 10/02/20 - 15/02/20**

- 2.1 Que es Java.
- 2.2 Versiones y ambiente de Java(JDK, IDE, máquina virtual, etc.).
- 2.3 Características de Java.
- 2.4 Tipos de programas en Java.
- 2.5 Componentes del lenguaje Java(variables, constantes, tipos de datos etc.).
- 2.6 Estructuras de control.
- 2.7 Arreglos.

### **3. Programación Estructurada 17/02/20 - 22/02/20**

- 3.1 Tipos de Variables.
- 3.2 Estructuras de Datos.
- 3.3 Estructuras de Control.
- 3.4 Funciones y Procedimientos.

### **4. Manejo de Excepciones 24/02/20 - 29/02/20**

- 4.1 Comprensión de excepciones y el manejo de estas.
- 4.2 Sentencia Try Catch.
- 4.3 Try anidados.

### **5. Paradigma de Objetos y UML 02/03/20 - 07/03/20**

- 5.1 Introducción al Paradigma Orientado a Objetos.
- 5.2 Introducción a UML.
- 5.3 Definiciones.
- 5.4 Simbología.

### **6. Programación Orientada a Objetos en Java(POO)**

**09/03/20 - 14/03/20**

- 6.1 Conceptos de la programación orientada a objetos.
- 6.2 Creación de clases y objetos.
- 6.3 Manejo de memoria.
- 6.4 Casteo de datos.
- 6.5 Constructores.
- 6.6 Herencia.
- 6.7 Polimorfismo.

### **7. Introducción a AWT y SWING 16/03/20 - 21/03/20 7.1**

- Frames.
- 7.2 Botones.
- 7.3 Textbox, etc..
- 7.4 Ejemplos.

## **8. Clases abstractas, interfaces y paquetes 23/03/20 - 28/03/20**

- 8.1 Control de acceso a clases y métodos.
- 8.2 Clases abstractas y métodos.
- 8.3 Interfaces.
- 8.4 Paquetes.

## **9. Manejo de hilos en Java 13/04/20 - 19/04/20**

- 9.1 Creación de hilos.
- 9.2 Multi-hilo.
- 9.3 Animación empleando hilos.

## **10. Archivos 20/04/20 - 25/04/20**

- 10.1 Introducción a flujo de datos.
- 10.2 Flujo de entrada y salida de datos.
- 10.3 Abrir, Cerrar y Modificar Archivos.
- 10.4 Buffer de lectura y escritura.

## **11. Listas y Colas 27/04/20 - 02/05/20**

- 11.1 Pilas usando listas.
- 11.2 Colas usando listas.
- 11.3 Listas n-encadenadas.

### **PUNTOS IMPORTANTES A CONSIDERAR:**

- Para tener derecho a nota de laboratorio se debe cumplir con el 80% de asistencia a clase de laboratorio a excepción de presentar carta de trabajo membretada.
- No se aceptarán entregas tarde sobre tareas, practicas, exámenes cortos, exámenes finales y proyectos. Si se llegaran a aceptar se tendrá una penalización del 30% sobre la nota obtenida sin excepciones.
- El único medio de entrega de actividades será por medio de Classroom.

### **BIBLIOGRAFIA:**

- JOYANES, L. y ZAHONERO, I. "Programación en Java 2 (algoritmos, estructura de datos y Programación Orientada a Objetos)". España, McGraw-Hill / Interamericana de España, S. A. 2002, PP 725
- BUDD, Timothy. "Introducción a la programación orientada a objetos", EUA, AddisonWesley, Iberoamericana, S. A. 1994, P. 409
- JOYANES, L. "Programación en Turbo Pascal Versiones 5.5, 6.0, y 7.0", (2da Edición), México, McGraw-Hill / Interamericana de España, S. A. 1995, PP. 914.
- Manuales de Referencia de Java.
- Cualquier otro material (escrito o digital) entregado en clase.