

**Programa de Curso**

UNIVERSIDAD NACIONAL

SEDE REGIONAL BRUNCA

CAMPUS PÉREZ ZELEDÓN

INGENIERÍA EN SISTEMAS DE INFORMACIÓN – BA-INFORM, 572101



<b>INFORMACIÓN CURSO</b>	
<b>NOMBRE DEL CURSO:</b>	<b>Programación I</b>
<b>NRC:</b>	<b>50099</b>
<b>TIPO DE CURSO:</b>	<b>Regular</b>
<b>CÓDIGO DE CURSO:</b>	<b>EIF-201</b>
<b>NIVEL Y GRADO ACADÉMICO:</b>	<b>I Nivel - Bachillerato</b>
<b>PERIODO LECTIVO:</b>	<b>II Ciclo 2024 (17 Semanas)</b>
<b>MODALIDAD:</b>	<b>Presencial</b>
<b>NATURALEZA:</b>	<b>Teórico-Práctico</b>
<b>CRÉDITOS:</b>	<b>4</b>
<b>HORAS TOTALES SEMANALES:</b>	<b>10</b>
<b>HORAS DEL CURSO:</b>	<b>Teoría: 2, Práctica: 2, Estudio Independiente: 6.</b>
<b>HORARIO:</b>	<b>Miércoles 08:00 a.m. – 12:00 p.m.</b>
<b>HORARIO DE ATENCIÓN ESTUDIANTE:</b>	<b>Miércoles 15:00 p.m. – 16:00 p.m</b>
<b>REQUISITOS:</b>	<b>EIF-200: Fundamentos de Informática MAT-030: Matemática para Informática I</b>
<b>CORREQUISITOS:</b>	<b>Ninguno</b>
<b>PERSONA DOCENTE:</b>	<b>Msc Alejandro Flores Quesada</b>
<b>GUIA ACADÉMICO:</b>	<b>Msc Alejandro Flores Quesada</b>
<b>AULA:</b>	<b>L001</b>
<b>GRUPO:</b>	<b>80</b>

En esta Universidad nos comprometemos a prevenir, investigar y sancionar el hostigamiento sexual entendido como toda conducta o comportamiento físico, verbal, no verbal escrito, expreso o implícito, de connotación sexual, no deseado o molesto para quien o quienes lo reciben, reiterado o aislado. Si usted está siendo víctima de hostigamiento diríjase a la Fiscalía de Hostigamiento Sexual de la UNA o llame al teléfono: 2277-3961.

La Universidad Nacional en el Estatuto Orgánico tiene como valor fundamental promover la originalidad, el pensamiento crítico y la excelencia académica. Por tal motivo se prohíbe utilizar inteligencia artificial (IA) en el desarrollo de trabajos académicos a no ser que el profesor indique un procedimiento adecuado para hacer uso de esta. Esta instrucción tiene como objetivo principal promover la adquisición de conocimientos mediante el esfuerzo propio, evitando el plagio o la copia.

## I. Descripción

Este curso le da continuidad al curso EIF 200 Fundamentos de Informática, en el aprendizaje por parte de los estudiantes, de los aspectos básicos del diseño orientado a objetos y su respectiva programación, particularmente usando el lenguaje C++. En este curso el estudiante aplicará técnicas de desarrollo de software.

De esta forma el curso de Programación I profundiza la temática desarrollada en el curso anterior, principalmente introduciendo las relaciones entre clases y objetos, colecciones y persistencia de los datos, los cuales le dan al estudiante una perspectiva más completa de la programación de aplicaciones en el paradigma de orientación a objetos, estableciendo las bases para curso siguiente.

## II. OBJETIVOS

### a. Objetivo general

Diseñar e implementar a nivel básico con base en el paradigma de programación orientada a objetos, las principales técnicas de programación y estructuras de datos que permiten al estudiante desarrollar las habilidades necesarias para el desarrollo de software de pequeña a mediana complejidad.

### b. Objetivos específicos

1. Identificar y clasificar los conceptos más importantes mediante el paradigma de orientación a objetos para definir la arquitectura de un programa.
2. Analizar y proponer soluciones computacionales a problemas mediante un proceso de descomposición y especialización.
3. Diseñar e implementar soluciones computacionales a problemas a partir del paradigma de orientación a objetos en lenguaje C++.
4. Analizar y discernir adecuadamente las estructuras de programación que permiten resolver un problema, disponibles en el lenguaje de programación C++.
5. Indagar de forma autodidacta temas relacionados con el curso.
6. Propiciar el desarrollo de habilidades para el trabajo en equipo: liderazgo efectivo, colaboración, división efectiva y adecuada de tareas, compromiso con el propio aprendizaje, resolución de conflictos, etc.



### c. Habilidades y destrezas

A través del curso se pretende facilitar y promover el desarrollo de al menos las siguientes competencias del estudiante:

- ✓ Identificar y clasificar conceptos a través del paradigma de orientación a objetos para definir la arquitectura de un programa que da solución a un problema.
- ✓ Analizar problemas y plantear su solución mediante un proceso de descomposición y especialización.
- ✓ Analizar y discernir adecuadamente las estructuras disponibles en el lenguaje de programación C++ para la solución de problemas.
- ✓ Diseñar soluciones a problemas usando el paradigma de POO.
- ✓ Implementar aplicaciones básicas en el paradigma de orientación a objetos en lenguaje C++.



### III. Contenidos/Aprendizajes Integrales:

#### a. Tema 1. Conceptos básicos de programación basado en objetos.

- i. Repaso de sentencia condicional (if), sentencia de selección múltiple (switch), sentencias de control de flujo iterativa (ciclos), vectores (arreglos), matrices (arreglos bidimensionales).
- ii. Repaso de conceptos tales como: Clase, objeto, estado, identidad, atributos, métodos, visibilidad, vista, abstracción, encapsulación, calificador const.

#### b. Tema 2. Uso de memoria y uso de punteros.

- i. Tipos de memoria
- ii. Asignación de memoria de las instancias
- iii. Liberación de memoria de las instancias (destructor virtual)
- iv. Uso de punteros

#### c. Tema 3. Colecciones básicas

- i. Arreglos de tipos básicos y de objetos (dinámicos y estáticos), administración de operaciones básicas.
  1. Unidimensional
  2. Bidimensional
- ii. Lista simplemente enlazada, administración de operaciones básicas.
  1. Creación, ingreso, eliminación, sustitución de elementos en una lista.

#### d. Tema 4. Relaciones. (Diseño e implementación)

- i. Representación e implementación de cada una de las relaciones entre clases: dependencia, asociación, agregación y composición, su cardinalidad y navegabilidad, dependencia mutua.
- ii. Herencia (Especialización y Generalización).
  1. Invocación de constructores y destructores
  2. Enlace dinámico
- iii. Diagramación de clases
  1. Estándar de diagramas de clases UML
  2. Visibilidad de los miembros de la clase
  3. Tipos de datos del diagrama de clases UML

4. Navegabilidad
5. Principios, leyes y conceptos Multiplicidad

**e. Tema 5. Concepto de polimorfismo.**

- i. Sobrecarga de métodos para extensión y sustitución (overwriting).
- ii. Sentencia virtual, métodos virtuales puros, clases abstractas.
- iii. Enlace dinámico
- iv. Miembros de instancia y miembros de clase (static)

**f. Tema 6. Persistencia básica de los datos (almacenamiento en archivos de texto)**

- i. Almacenamiento de los datos (por medio de funciones o métodos)
- ii. Recuperación de los datos en memoria (por medio de funciones o métodos)
- iii. Estructura de los datos en el archivo
- iv. Uso del getline
- v. Uso del stringstream



#### IV. Metodología

Para el logro de los objetivos se propone un abordaje de los contenidos de manera magistral por parte del profesor, donde el estudiante se sienta acompañado en todo momento. Para esto, se espera del estudiante una intensa participación en todo el proceso, como persona comprometida con su propio aprendizaje, aprovechando la presencialidad total de sus profesores en el aula o laboratorio.

- Se desarrollarán ejercicios prácticos (presenciales o vídeo) asociados con los temas. Se realizarán prácticas y laboratorios semanales que le permitan al estudiante en su casa, practicar los conceptos teóricos guiado por el profesor y continuar con el proceso en su tiempo de estudio independiente. Se realizarán diversas actividades como quices, tareas, laboratorios, trabajo a concluir y proyectos programados con el fin de promover la asimilación de los conceptos.
- Se promoverá la investigación constructiva en los estudiantes, de tal manera que estos brinden aportes al aprendizaje del educando y construyan sus propias soluciones a los problemas planteados.
- El ambiente de desarrollo (IDE) a utilizar durante el curso será VS Community – C++.
- El material entregado en clase puede estar en idioma español o inglés, es competencia importante del estudiante poder utilizar el conocimiento que tiene en esta materia también.
- Para la realización de este curso se requiere contar con una cuenta gratuita en github.com, realizada con el correo institucional de cada estudiante, y utilizando el nombre y apellidos para poder identificarlo fácilmente.

Por ser un curso teórico-práctico la asistencia es **necesaria**.

## V. Evaluación

EVALUACIÓN	
Descripción	Porcentaje
<b>Exámenes Cortos (5)</b>	
Los quices corresponden a pruebas cortas, comprobación de lecturas y/o ejercicios de desarrollo que deberán completarse en forma <b>individual</b> dentro de las horas de clase. (3% cada quiz).	<b>15%</b>
<b>Examen (2)</b>	
Los exámenes buscan medir y evaluar la comprensión de cada estudiante del material estudiado durante el curso y del trabajo realizado en los proyectos. Los exámenes deben realizarse y entregarse <b>individualmente</b> . (25% cada examen)	<b>50%</b>
<b>Proyecto (2)</b>	
Los proyectos servirán para evaluar aspectos prácticos concretos de los temas estudiados en el curso. Los enunciados serán entregados con anticipación, con una descripción detallada y con su respectiva tabla de calificación. Asimismo, cada proyecto será desarrollado de forma <b>individual</b> . (15% para el primer proyecto y un 20% en el segundo proyecto)	<b>35%</b>
<b>TOTAL</b>	<b>100%</b>



## Rúbricas para evaluación de proyectos

Rúbrica para evaluación del proyecto #1		
Rubro	Descripción	Puntos
Originalidad	Evalúa la creatividad y la innovación en la solución propuesta.	4
	- Solución estándar o sin innovación: 0 puntos	
	- Solución con algunas ideas creativas: 2 puntos	
	- Solución altamente creativa e innovadora: 4 puntos	
Calidad del Código Fuente	Evalúa la legibilidad, organización y buenas prácticas de programación.	5
	- Código confuso, mal organizado y sin comentarios: 0 puntos	
	- Código legible, bien estructurado con comentarios adecuados: 2 puntos	
	- Código muy legible, modular, bien documentado y con buenas prácticas: 5 puntos	
Compleitud del Programa	Evalúa si el programa cumple con todos los requisitos funcionales especificados.	3
	- Programa incompleto o con funcionalidades faltantes: 0 puntos	
	- Programa funcional con algunas características incompletas: 1 punto	
	- Programa completo y funcional según lo especificado: 3 puntos	
Ejecución	Evalúa la ejecución correcta del programa, manejo de errores y eficiencia.	3
	- Errores frecuentes y problemas de ejecución: 0 puntos	
	- Ejecución correcta con algunos errores menores: 1 punto	
	- Ejecución sin errores significativos y eficiente: 3 puntos	
<b>TOTAL</b>		<b>15%</b>



Rúbrica para evaluación del proyecto #2		
Rubro	Descripción	Puntos
Originalidad	Evalúa la creatividad y la innovación en la solución propuesta.	5
	- Solución estándar o sin innovación: 0 puntos	
	- Solución con algunas ideas creativas: 2 puntos	
	- Solución altamente creativa e innovadora: 5 puntos	
Calidad del Código Fuente	Evalúa la legibilidad, organización y buenas prácticas de programación.	7
	- Código confuso, mal organizado y sin comentarios: 0 puntos	
	- Código legible, bien estructurado con comentarios adecuados: 3 puntos	
	- Código muy legible, modular, bien documentado y con buenas prácticas: 7 puntos	
Compleitud del Programa	Evalúa si el programa cumple con todos los requisitos funcionales especificados.	4
	- Programa incompleto o con funcionalidades faltantes: 0 puntos	
	- Programa funcional con algunas características incompletas: 2 punto	
	- Programa completo y funcional según lo especificado: 4 puntos	
Ejecución	Evalúa la ejecución correcta del programa, manejo de errores y eficiencia.	4
	- Errores frecuentes y problemas de ejecución: 0 puntos	
	- Ejecución correcta con algunos errores menores: 2 punto	
	- Ejecución sin errores significativos y eficiente: 4 puntos	
TOTAL		20%

## Entrega de Trabajos

### - Entrega en el Aula Virtual:

Todos los trabajos deben ser entregados (en el Aula Virtual) el día, hora y bajo las condiciones que indique el profesor. Si no se entrega a tiempo el trabajo, se perderá la totalidad del valor del mismo.

Adicionalmente se compartirá en GitHub los proyectos a desarrollar durante el curso obligatoriamente al académico responsable, por medio de la cuenta. (alejandrofq).

### - Forma de Entrega:

Los proyectos deberán ser entregados de forma digital, tanto el documento escrito como la presentación del trabajo. Ambos identificados con el nombre en un archivo comprimido. Serán entregados por medio del aula virtual. <https://www.aulavirtual.una.ac.cr/>

En vista de la nueva tendencia de la conservación del ambiente, ningún trabajo será impreso. Todo trabajo será enviado o entregado de forma digital al profesor por medio del aula virtual. <https://www.aulavirtual.una.ac.cr/>

### - No se recibirán trabajos al correo electrónico:

No se recibirán trabajos (investigaciones, proyecto, laboratorios) al correo electrónico, a salvedad de que no esté en funcionamiento el aula virtual.

### - Trabajo Individual:

Todos los trabajos por entregar por parte del estudiante serán realizados de forma individual.

## **Evaluación y Justificación de Ausencias**

### **- Naturaleza Evaluativa:**

Debido a la naturaleza evaluativa de las prácticas de laboratorio, exámenes cortos, presentación de investigaciones y avances del proyecto, la ausencia a estas deberá justificarse según normativa universitaria respectiva.

### **- Evaluación del Curso:**

Por la naturaleza de los contenidos del curso, los contenidos desarrollados son acumulativos para los laboratorios y exámenes cortos. Al ser un curso cuya evaluación contempla aspectos que se desarrollan a lo largo del curso, como los exámenes cortos, exámenes y proyectos programados, el curso no tiene examen extraordinario. La suma de los porcentajes obtenidos por el estudiante en los rubros anteriores determina su nota de aprovechamiento (NA). Si esta es superior o igual a 70%, el estudiante aprueba el curso, y si la NA es menor a 70%, el estudiante reprueba el curso.

La no presentación del proyecto será motivo para que el estudiante repruebe el curso.

## **Comunicación**

### **- Correo Electrónico:**

Toda comunicación será realizada en primera instancia por medio de correo electrónico desde el aula virtual. Es responsabilidad del estudiante tener actualizado su correo electrónico en dicha plataforma.

### **- Bot Informativo:**

Como parte de la comunicación entre el grupo de estudiantes y profesor, se utilizará el BOT INFORMATIVO SRB (Sr.Bot) para llevar e informar en tiempo real las notas obtenidas por parte del grupo de estudiantes. Durante el primer día de lecciones, el profesor se encargará de explicar el funcionamiento del Sr.Bot.

### **- Grupo de Telegram:**

Adicionalmente, el profesor creará un grupo de comunicación en la plataforma Whatsapp con el fin de alertar de forma inmediata algún acontecimiento, noticia, actualización de contenido o tarea requerida. Es necesario que cada estudiante envíe un mensaje a los administradores definidos en clase para ser incluidos.

## **Reglamento y Normativa**

### **- Detalle de Evaluaciones:**

Se adjunta en el Anexo 1, el detalle de los aspectos a evaluar en cada uno de los proyectos y laboratorios. El plagio y la copia no se permiten y serán sancionados según lo establecen los reglamentos de la institución. Ver artículos 24 y 25 del Reglamento General sobre los Procesos de Enseñanza Aprendizaje.





### - Artículo 24. Plagio:

Se considera plagio la reproducción parcial o total de documentos ajenos presentándolos como propios. En el caso que se compruebe el plagio por parte del estudiante, perderá el curso. Si reincide, será suspendido de la carrera por un ciclo lectivo, y si la situación se repite una vez más, será expulsado de la Universidad.

### - Artículo 25. Copia:

Se considera copia todo documento o medio no autorizado utilizado de manera subrepticia por el estudiante durante una prueba evaluativa. De comprobarse la copia en la realización de una evaluación, esta será calificada con nota de cero y el estudiante perderá el porcentaje correspondiente a esa evaluación, independientemente de la eventual sanción disciplinaria establecida en la normativa institucional.



### VI. Cronograma

El tiempo asignado a cada tema podría ser modificado según sea necesario, y las fechas de cada evaluación se establecerán en el cronograma del curso. Una vez establecidas las fechas de entrega o realización para cada evaluación, estas no podrán variarse salvo casos de fuerza mayor. Cuando se defina una fecha de entrega en el aula virtual del curso, se entiende que esta es el límite máximo para dicha entrega. Salvo que existan circunstancias extraordinarias, no se ampliarán los plazos de entrega de ninguna evaluación. Es responsabilidad de cada estudiante verificar las fechas de entrega en la sección correspondiente del aula virtual.

CRONOGRAMA				
# Sesión	Fecha	Aprendizajes integrales	Actividades	Recursos didácticos
1	24 JUL 24	<ul style="list-style-type: none"> <li>Entrega y aprobación programa curso.</li> <li>Explicación del Aula Virtual.</li> <li>Explicación del Sr.Bot.</li> <li>Repaso de conceptos</li> <li>Introducción a las Buenas Prácticas (Uso de espacio de nombres, guardas, definición de clases, etc.)</li> <li>Aplicación Diagnóstico</li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> <li>Computadora.</li> <li>Aula Virtual.</li> </ul>
2	31 JUL 24	<ul style="list-style-type: none"> <li><b>Entrega Enunciado Proyecto #1.</b></li> <li>Arreglos y matrices</li> <li>Guía para el uso de GIT</li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> <li>Práctica – Resolución de ejercicios.</li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> <li>Computadora.</li> <li>Aula Virtual.</li> </ul>
3	07 AGO 24	<ul style="list-style-type: none"> <li>Arreglos y matrices</li> <li>Punteros</li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> <li>Práctica – Resolución de ejercicios.</li> <li><b>Examen Corto #1.</b></li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> <li>Computadora.</li> <li>Aula Virtual.</li> </ul>
4	14 AGO 24	<ul style="list-style-type: none"> <li>Punteros</li> <li>Arreglos de punteros dinámicos</li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> <li>Práctica – Resolución de ejercicios.</li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> <li>Computadora.</li> <li>Aula Virtual.</li> </ul>
5	21 AGO 24	<ul style="list-style-type: none"> <li>Principios básicos de programación orientada a objetos.</li> <li>Introducción al POO (Clase, Objeto, Estado, Identidad, Atributos)</li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> <li>Práctica – Resolución de ejercicios.</li> <li><b>Examen Corto #2.</b></li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> <li>Computadora.</li> <li>Aula Virtual.</li> </ul>
6	28 AGO 24	<ul style="list-style-type: none"> <li>Polimorfismo, Clases abstractas</li> <li><b>Primer Avance Proyecto #1 (Commit)</b></li> </ul>	<ul style="list-style-type: none"> <li>Presencialidad.</li> <li>Práctica – Resolución de</li> </ul>	<ul style="list-style-type: none"> <li>Pizarra Inteligente.</li> </ul>



			ejercicios.	❖ Computadora. ❖ Aula Virtual. ❖ GitHub.
7	04 SET 24	❖ <b>Examen #1</b>	❖ Presencialidad. ❖ <b>Aplicación Examen #1.</b>	❖ Papel y lápiz.
8	11 SET 24	❖ <b>Entrega, presentación y defensa del Proyecto#1.</b> ❖ <b>Entrega Enunciado Proyecto #2.</b>	❖ Presencialidad. ❖ <b>Revisión Proyecto #1.</b>	❖ Computadora. ❖ Aula Virtual. ❖ GitHub.
9	18 SET 24	❖ Listas, administración de operaciones básicas	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios.	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual.
10	25 SET 24	❖ Listas ❖ Manejo de archivos, administración de operaciones básicas	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios. ❖ <b>Examen Corto #3.</b>	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual.
	02 OCT 24	❖ Semana Universitaria Campus PZ. ❖ Listas	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios.	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual.
12	09 OCT 24	❖ Listas ❖ Manejo de archivos, administración de operaciones básicas ❖ <b>Primer Avance Proyecto #2 (Commit)</b>	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios. ❖ <b>Examen Corto #4.</b>	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual. ❖ GitHub.
13	16 OCT 24	❖ Semana Universitaria Campus Coto. ❖ Listas ❖ Manejo de archivos	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios.	❖ Computadora. ❖ Aula Virtual.
14	23 OCT 24	❖ Listas ❖ Manejo de archivos	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios. ❖ <b>Examen Corto #5.</b>	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual.
15	30 OCT 24	❖ Recursividad y peso de algoritmos ❖ <b>Segundo Avance Proyecto #2 (Commit)</b>	❖ Presencialidad. ❖ Práctica – Resolución de ejercicios.	❖ Pizarra Inteligente. ❖ Computadora. ❖ Aula Virtual. ❖ GitHub.
16	06 NOV 24	❖ <b>Examen #2.</b>	❖ Presencialidad. ❖ <b>Aplicación Examen #2.</b>	❖ Papel y lápiz.
17	13 NOV 24	❖ <b>Entrega, presentación y defensa del Proyecto #2.</b>	❖ Presencialidad. ❖ <b>Revisión Proyecto #2.</b>	❖ Computadora. ❖ Aula Virtual. ❖ GitHub.
	20 NOV 24	❖ <b>Entrega Promedios Finales</b>	❖ <b>Promedios.</b>	❖ <b>Aula Virtual.</b> ❖ <b>Sr.Bot</b>





## VII. Recursos Bibliográficos

Deitel, Harvey y Deitel Paul. (2016). Como programar en C++. Prentice Hall. 11 Edición. México.

Joyanes, Luis. (1996). Programación Orientada a objetos: conceptos, modelado, diseño y codificación en C++. McGraw Hill. 1 Edición, España.

Langsam Yedidyah, Augenstein Moshe, Tenenbaum Aaron. (1997). Estructuras de Datos con C y C++. Pearson Education, 2da Edición. USA.

Savitech, Walter. (2000) Resolución de problemas con C++. Prentice Hall, 1 Edición.

Schildt, Herbert. (1995). C++ Guía de Auto enseñanza. Osborne/McGraw-Hill, Primera edición.

\* Bases de datos en la biblioteca de la UNA (vía web): ACM (Association for Computing Machinery) y EBSCO:Computers & Applied Sciences Complete.

\* Compendio de material bibliográfico seleccionado por el profesor.

## Lectura y Aprobación del Programa de Curso

**Se presenta, lee y se discute el programa del curso: Programación I**

**NRC: 50099**

**Grupo: 80**

[illegible]




Firma Académico:

Msc. Alejandro Flores Quesada