



| Nombre del Cuso: Introducción a la Programación y Computación II  |   |
|---|---|
| <b>Código:</b> 771  | <b>Créditos:</b> 5  |
| <b>Escuela:</b> CIENCIAS Y SISTEMAS   | <b>Área a la que pertenece:</b> Programación  |
| <b>Pre requisito:</b> Introducción a la Programación y Computación I (770)<br>Matemática Intermedia (107)<br>Lógica Matemática (795)<br>Matemática de Computo 1 (960) | <b>Post requisito:</b> Organización Computacional (964)<br>Estructura de Datos (772)<br>Org. Lenguajes y Compiladores 1 (777) |
| <b>Categoría:</b> Obligatorio   | <b>Semestre:</b> 2do. Semestre 2020   |
| <b>Docente:</b>   | <b>Auxiliar:</b>  |
| <b>Edificio:</b> -  | <b>Sección:</b>   |
| <b>Salón del curso:</b> meet  | <b>Salón de laboratorio:</b> meet   |
| <b>Horas por semana del curso:</b> 4  | <b>Horas por semana del laboratorio:</b> 2  |
| <b>Días que se imparte el curso:</b> Jueves y viernes   | <b>Días que se imparte el laboratorio:</b>  |
| <b>Horario del curso:</b> 07:10 - 08:50   | <b>Horario del laboratorio:</b>   |

### 1. Descripción del curso

Este curso está diseñado para que el estudiante inicie el proceso de modelado de sistemas de software utilizando los conceptos de la programación orientada a objetos y los diagramas que el lenguaje unificado de datos proporciona. También se introducirá en los conceptos de bases de datos relacionales y la importancia de la información en el contexto empresarial.

### 2. Objetivos

#### General

Preparar al estudiante para desarrollar aplicaciones de software utilizando un enfoque orientado a objetos.

#### Específicos

1. Modelar problemas de forma estándar y profesional.
2. Entender metodologías de desarrollo para construir aplicaciones de software.
3. Estructurar datos con base al modelado entidad relación.



2

0

2

0

S  
E  
M  
E  
S  
T  
R  
E

2

### 3. Metodología

1. El curso se impartirá a través de clases magistrales **virtuales** dos días por semana, con duración de dos periodos cada día.
2. El laboratorio se impartirá **de manera virtual** una vez por semana, con duración de dos períodos cada día.
3. Durante el semestre, se asigna un proyecto de desarrollo de software dividido en tres fases, a realizarse de manera individual; así como tareas, ejercicios y pruebas cortas.

### 4. Competencias terminales

Al finalizar el curso el estudiante desarrolla las siguientes competencias:

- Dominio en el modelado relacional.
- Capacidad para implementar modelos bajo el lenguaje de consulta estructurado -SQL-
- Capacidad para aplicar metodologías de programación y desarrollo de aplicaciones de software.
- Dominio en el modelado UML.

### 5. Observaciones

1. Es obligatorio acumular el 80% de asistencia antes de cada parcial (de lo contrario no se tendrá derecho a examen).
2. El laboratorio se calificará sobre 100, y será equivalente a 30 puntos de zona.
3. El proyecto se hará en tres fases, cada fase tendrá dos entregables.
4. El catedrático revisará las notas obtenidas en el curso y el laboratorio. Podrá decidir sí es necesaria una segunda revisión a cada fase de los proyectos o práctica y considerar nuevamente la ponderación obtenida en cada fase del proyecto o práctica.
5. Las notas de cada proyecto serán publicadas por el catedrático del curso en el transcurso del semestre, el estudiante tendrá 8 días como máximo para pedir revisión de proyecto.
6. El laboratorio debe aprobarse con 61 puntos.
7. Es obligatorio ganar el laboratorio para tener derecho a evaluación final del curso.
8. No habrá proyecto de retrasada, ni reposición de nota de laboratorio.
9. El curso se aprueba con 61 puntos.



2

0

2

0

SEMESTRE

2

**6. Contenido temático del curso**

| Unidad  | Tema  |
|---|---|
| <p><b>1. Bases de datos relacionales</b></p>                                | <p><b>1. Bases de datos relacionales</b></p> <p><b>1.1. Un lenguaje común</b></p> <p>1.1.1. ¿Qué es un dato?</p> <p>1.1.2. ¿Qué es una base de datos?</p> <p>1.1.3. ¿Qué es un modelo de datos?</p> <p><b>1.2. El Modelo Relacional</b></p> <p>1.2.1. Diagrama Entidad Relación</p> <p>1.2.1.1. ¿Qué es una entidad?</p> <p>1.2.1.2. ¿Qué es una tupla?</p> <p>1.2.1.3. Atributos</p> <p>1.2.2. Relaciones básicas</p> <p>1.2.2.1. De uno a uno</p> <p>1.2.2.2. De uno a muchos</p> <p>1.2.2.3. De muchos a muchos</p> <p>1.2.3. Identificadores</p> <p>1.2.3.1. Llaves primarias (simples y compuestas)</p> <p>1.2.3.2. Llaves foráneas</p> <p><b>1.3. Mapeo físico de una base de datos relacional</b></p> <p>1.3.1. Introducción a SQL</p> <p>1.3.1.1. Data Definition Language (DDL)</p> <p>1.3.1.2. Data Manipulation Language (DML)</p> <p>1.3.1.3. Constraints (para PK y FK)</p> <p><b>1.4. Manejadores de base de Datos</b></p> <p><b>1.5. Sentencias básicas</b></p> <p>1.5.1. CREATE</p> <p>1.5.2. INSERT</p> <p>1.5.3. SELECT / WHERE / ORDER BY / GROUP BY</p> <p>1.5.4. UPDATE</p> <p>1.5.5. JOIN</p> |
| <p><b>2. Metodologías para el desarrollo de sistemas de información</b></p> | <p><b>2. Metodologías para el desarrollo de sistemas de información</b></p> <p><b>2.1. ¿Qué es un sistema de información?</b></p> <p><b>2.2. Arquitectura de un sistema de información</b></p> <p>2.2.1. Presentación</p> <p>2.2.2. Lógica del negocio</p> <p>2.2.3. Datos</p> <p><b>2.3. Modelado de sistemas</b></p> <p>2.3.1. Abstracción</p> <p>2.3.2. Conceptualización</p> <p>2.3.3. Simbolización</p> <p><b>2.4. Metodología para desarrollar sistemas de información</b></p> <p><b>2.5. ¿Cómo se construye un sistema de información?</b></p> <p><b>2.6. ¿Cuál es el ciclo de vida para el desarrollo de sistemas de información?</b></p>   |



|   |   |
|---|---|
|   | <ul style="list-style-type: none"> <li>2.6.1. Toma de requerimientos</li> <li>2.6.2. Análisis y Diseño</li> <li>2.6.3. Desarrollo del software</li> <li>2.6.4. Implementación</li> <li>2.6.5. Pruebas</li> <li>2.6.6. Documentación</li> <li><b>2.7. Metodología de desarrollo ágil</b> <ul style="list-style-type: none"> <li>2.7.1. ¿Qué es la metodología ágil?</li> <li>2.7.2. Beneficios del método ágil</li> <li>2.7.3. Metodologías ágiles</li> <li>2.7.4. Roles en el método ágil</li> <li>2.7.5. Prácticas ágiles</li> </ul> </li> </ul>   |
| <p><b>3. Modelado Unificado con UML</b></p> | <ul style="list-style-type: none"> <li><b>3. Lenguaje de Modelado Unificado (UML)</b> <ul style="list-style-type: none"> <li><b>3.1. Lenguaje de modelado</b></li> <li><b>3.2. Introducción a UML</b></li> <li><b>3.3. Herramientas de modelado</b></li> <li><b>3.4. Modelado con Diagrama de Casos de Uso</b> <ul style="list-style-type: none"> <li>3.4.1. ¿Qué es un requisito?</li> <li>3.4.2. Tipos de requisito                             <ul style="list-style-type: none"> <li>3.4.2.1. Funcionales</li> <li>3.4.2.2. No Funcionales</li> </ul> </li> <li>3.4.3. Casos de Uso                             <ul style="list-style-type: none"> <li>3.4.3.1. ¿Qué es un caso de uso?</li> <li>3.4.3.2. Actores</li> <li>3.4.3.3. Tipos de Relación                                     <ul style="list-style-type: none"> <li>3.4.3.3.1. Extiende</li> <li>3.4.3.3.2. Incluye</li> <li>3.4.3.3.3. Generaliza</li> </ul> </li> <li>3.4.3.4. Diagrama de casos de uso</li> <li>3.4.3.5. Especificación de alto nivel</li> <li>3.4.3.6. Especificación expandida</li> </ul> </li> </ul> </li> <li><b>3.5. Modelo conceptual</b> <ul style="list-style-type: none"> <li>3.5.1. Introducción a las clases de negocio</li> <li>3.5.2. Atributos</li> <li>3.5.3. Relaciones</li> <li>3.5.4. Estrategias para identificar clases de negocio</li> <li>3.5.5. Construcción de un modelo conceptual</li> </ul> </li> <li><b>3.6. Modelado con Diagrama de Clases</b> <ul style="list-style-type: none"> <li>3.6.1. Clases</li> <li>3.6.2. Notación de clases</li> <li>3.6.3. Atributos</li> <li>3.6.4. Métodos</li> <li>3.6.5. Tipos de visibilidad (Atributos y Métodos)</li> <li>3.6.6. Relaciones de Clase</li> <li>3.6.7.                             <ul style="list-style-type: none"> <li>3.6.7.1. Asociación</li> <li>3.6.7.2. Realización</li> </ul> </li> </ul> </li> </ul> </li> </ul> |



2

0

2

0

S  
E  
M  
E  
S  
T  
R  
E

2

3.6.7.3. Agregación

3.6.7.4. Composición

**3.7. Modelado de Diagrama de Secuencia**

3.7.1. Línea de vida

3.7.2. Mensajes

3.7.3. Ejecución

3.7.4. Mensajes Self

3.7.5. Destrucción de un objeto

3.7.6. Comportamiento del sistema

3.7.7. Eventos y operaciones de un sistema

**3.8. Modelado con Diagrama de Estados**

3.8.1. Introducción a los diagramas de estado

3.8.1.1. Estado

3.8.1.2. Eventos

3.8.1.3. Envío de mensajes

3.8.2. Utilidad de los diagramas de estado para los casos de uso

3.8.3. Diagramas de estado del sistema

3.8.4. Tipos independientes y dependientes del estado

3.8.5. Notación de los diagramas de estado

3.8.5.1. Acciones de transición

3.8.5.2. Condiciones protectoras de las transiciones

3.8.5.3. Estados anidados

**3.9. Modelado con Diagrama de Actividades**

3.9.1. Acciones

3.9.2. Actividades

3.9.3. Transiciones

3.9.4. División y unión

3.9.5. Calles/Transiciones

**3.10. Modelado de Diagrama de Componentes**

3.10.1. Representando Componentes

3.10.2. Interfaces requeridas

3.10.3. Tipos de componentes

3.10.4. Usos comunes

3.10.4.1. Modelado de código fuente

3.10.4.2. Modelado de una versión ejecutable y bibliotecas

3.10.4.3. Modelado de una base de datos física



## 7. Evaluación del rendimiento académico

Según el Reglamento General de Evaluación y Promoción del Estudiante de la Universidad de San Carlos de Guatemala, la zona tiene valor de 75 puntos, la nota mínima de promoción es de 61 puntos y la zona mínima para optar a examen final es de 36 puntos.

| Procedimiento de evaluación |                             | Ponderación    |
|-----------------------------|-----------------------------|----------------|
| Clase                       | Tareas y/o cortos           | 06 pts.        |
|                             | Primer parcial              | 13 pts.        |
|                             | Segundo parcial             | 13 pts.        |
|                             | Tercer parcial              | 13 pts.        |
| <b>Total de clase</b>       |                             | <b>45 pts.</b> |
| Laboratorio                 | Proyecto Fase I             | 07 pts.        |
|                             | Entregable 1: 1 pto.        |                |
|                             | Entregable 2: 6 pts.        |                |
|                             | Proyecto Fase II            | 11 pts.        |
|                             | Entregable 1: 1 pto.        |                |
|                             | Entregable 2: 10 pts.       |                |
| Laboratorio                 | Proyecto Fase III           | 12 pts.        |
|                             | Entregable 1: 1 pto.        |                |
|                             | Entregable 2: 11 pts.       |                |
|                             | <b>Total de laboratorio</b> | <b>30 pts.</b> |
| Zona                        | 75 pts.                     |                |
| Examen Final                | 25 pts.                     |                |
| Nota de promoción           | 100 pts.                    |                |

## 8. Cronograma de actividades

| Tema principal   | Contenido a desarrollar  | Fecha            |
|--|--|------------------|
| Bases de datos relacionales  | Un lenguaje común  | 30-julio         |
| El modelo relacional   | - Diagrama entidad relación  | 31-julio         |
| El modelo relacional   | - Relaciones básicas<br>- Identificadores                                    | 06-agosto        |
| El modelo relacional   | Ejemplos prácticos del modelado entidad relación                             | 07-agosto        |
| Mapeo físico de una base de datos relacional y Manejadores de bases de datos | - Introducción a SQL<br>- Conociendo los distintos motores de bases de datos | 13-agosto        |
| Sentencias básicas SQL   | - Sentencias básicas de SQL<br>- Aplicaciones prácticas de SQL               | 14-agosto        |
| <b>Primer parcial</b>  |  | <b>28-agosto</b> |



|  |  |               |
|--|--|---------------|
| Metodologías para el desarrollo de sistemas de información | <ul style="list-style-type: none"> <li>- ¿Qué es un sistema de información?</li> <li>- Arquitectura de un sistema de información</li> <li>- Modelado de sistemas</li> <li>- Metodología para desarrollar sistemas de información</li> <li>- ¿Cómo se construye un sistema de información?</li> <li>- ¿Cuál es el ciclo de vida para el desarrollo de sistemas de información?</li> <li>- Métodos ágiles de desarrollo de software</li> </ul> | 20-agosto     |
| Lenguaje de Modelado Unificado                             | <ul style="list-style-type: none"> <li>- Lenguaje de modelado</li> <li>- Introducción a UML</li> <li>- Herramientas de modelado</li> </ul>   | 21-agosto     |
| Modelado de Diagrama de Casos de Uso                       | <ul style="list-style-type: none"> <li>- ¿Qué es un requisito?</li> <li>- Tipos de requisito</li> <li>- Casos de uso</li> </ul>  | 27-agosto     |
| Modelado de Diagrama de Casos de Uso                       | Aplicaciones prácticas de diagramas de casos de uso  | 03-septiembre |
| Modelado de Diagrama de Casos de Uso                       | Aplicaciones prácticas de diagramas de casos de uso  | 04-septiembre |
| Modelo Conceptual  | <ul style="list-style-type: none"> <li>- Introducción a las clases de negocio</li> <li>- Atributos</li> <li>- Relaciones</li> </ul>  | 10-septiembre |
| Modelo Conceptual  | <ul style="list-style-type: none"> <li>- Estrategias para identificar clases de negocio</li> <li>- Construcción de un modelo conceptual</li> </ul>   | 11-septiembre |
| Modelo Conceptual  | Aplicaciones prácticas del modelo conceptual   | 17-septiembre |
| Modelado de Diagrama de Clases                             | <ul style="list-style-type: none"> <li>- Clases</li> <li>- Notación de clases</li> <li>- Atributos</li> <li>- Métodos</li> <li>- Tipos de visibilidad</li> </ul>   | 18-septiembre |
| Modelado de Diagrama de Clases                             | Relaciones de clase  | 24-septiembre |
| Modelado de Diagrama de Clases                             | Aplicaciones prácticas de diagramas de clases  | 25-septiembre |
| <b>Semana de congreso estudiantil</b>                      | <b>28-septiembre al 3-octubre</b>  |               |
| <b>Segundo parcial</b>                                     | <b>09-octubre</b>  |               |
| Modelado de Diagrama de Secuencias                         | <ul style="list-style-type: none"> <li>- Línea de vida</li> <li>- Mensajes</li> <li>- Ejecución</li> <li>- Mensajes Self</li> <li>- Destrucción de un objeto</li> <li>- Comportamiento del sistema</li> <li>- Eventos y operaciones de un sistema</li> </ul>   | 08-octubre    |
| Modelado de Diagrama de Secuencias                         | Aplicaciones prácticas de diagramas de secuencias  | 15-octubre    |



|                                     |  |                     |
|-------------------------------------|--|---------------------|
| Modelado de Diagrama de Estados     | - Introducción<br>- Utilidad<br>- Diagramas de estado del sistema                            | 16-octubre          |
| Modelado de Diagrama de Estados     | - Tipos independientes y dependientes<br>- Notación  | 22-octubre          |
| Modelado de Diagrama de Actividades | - Acciones<br>- Actividades<br>- Transiciones<br>- División y unión<br>- Calles/Transiciones | 23-octubre          |
| Modelado de Diagrama de Actividades | Aplicaciones prácticas de diagramas de actividades   | 29-octubre          |
| Modelado de Diagrama de Componentes | - Representando Componentes<br>- Interfaces requeridas<br>- Tipos de componentes             | 30-octubre          |
| Modelado de Diagrama de Componentes | - Usos comunes<br>- Aplicaciones prácticas de diagramas de componentes                       | 05-noviembre        |
| <b>Tercer parcial</b>               |  | <b>06-Noviembre</b> |

## 9. Bibliografía

- Jiménez, C. Fundamentos de SQL. 3ra. Edición. McGraw Hill. 2009.
- Scrum Manager. Enlace [<http://www.scrummanager.net/ok>]. 2009.
- Craig Larman, UML y Patrones, Introducción al análisis y diseño orientado a objetos. Prentice Hall.