



A) CURSO

| Clave | Asignatura |
|-------|-----------------|
| 5708 | Programación II |

| Horas de teoría por semana | Horas de práctica por semana | Horas trabajo adicional estudiante | Créditos | Horas Totales |
|----------------------------|------------------------------|------------------------------------|----------|---------------|
| 4 | 0 | 4 | 8 | 64 |

B) DATOS BÁSICOS DEL CURSO

| | IEA | IM | IMA | IME | IMT |
|------------------------------|-----|----|-----|-----|-----------------------|
| Nivel: | | | | | V |
| Tipo (Optativa, Obligatoria) | | | | | Obligatoria |
| Prerrequisito: | | | | | Programación I (5707) |
| Clasificación CACEI: | | | | | CI |

C) Objetivo General del curso

Al finalizar el curso el estudiante será capaz de:

Desarrollar aplicaciones visuales utilizando lenguaje de programación Visual C++. Aplicar las directivas del preprocesador correspondientes en la creación de bibliotecas de código.



D) CONTENIDOS Y MÉTODOS POR UNIDADES Y TEMAS

| | | |
|---|--|-------------|
| 1. Introducción a la Programación Orientada a Objetos (POO) | | 8 h |
| Objetivo | Comprender la diferencia entre los paradigmas de programación estructurada y orientada a objetos. | |
| Específico: | Aplicar, en lo general, los conceptos de la programación orientada a objetos. | |
| 1.1 Objeto 1.2 Clase 1.3 Método 1.4 Mensaje 1.5 Herencia 1.6 Abstracción 1.7 Encapsulamiento 1.8 Polimorfismo 1.9 Recolección de basura | | |
| Lecturas y otros recursos | Se recomienda leer los temas de la bibliografía sugerida. | |
| Métodos de enseñanza | Se presentan los conceptos teóricos y posteriormente ejercicios de aplicación. | |
| Actividades de aprendizaje | Se realizan tareas y prácticas. | |
| 2. OOP con C++ | | 24 h |
| Objetivo | Desarrollar programas en lenguaje de programación C++. | |
| Específico: | | |
| 2.1 Introducción 2.2 Directivas del preprocesador 2.2.1 Creación de bibliotecas 2.3 Conceptos 2.3.1 Creación de clases 2.3.2 Constructores y destructores 2.3.3 Acceso a miembros 2.3.3.1 Modificadores de protección (private, public, protected) 2.4 Sobrecarga 2.5 Herencia 2.5.1 Clase base y clase derivada 2.5.2 Redefinición de miembros en clases derivadas 2.6 Polimorfismo 2.6.1 Funciones virtuales 2.6.2 Clases abstractas 2.7 Genericidad | | |
| Lecturas y otros recursos | Se recomienda leer los temas de la bibliografía sugerida. | |
| Métodos de enseñanza | Se presentan los conceptos teóricos y posteriormente ejercicios de aplicación. | |
| Actividades de aprendizaje | Se realizan tareas y prácticas. | |
| 3. Uso de Librerías de Plantillas Estándar (STL, por sus siglas en inglés) | | 8 h |
| Objetivo | Utilizar librerías específicas en la implementación de programas que caractericen soluciones particulares. | |
| Específico: | | |
| 3.1 Iteradores 3.2 Contenedores 3.2.1 List 3.2.2 Stack 3.2.3 Queue | | |
| Lecturas y otros recursos | Se recomienda leer los temas de la bibliografía sugerida. | |
| Métodos de enseñanza | Se presentan los conceptos teóricos y posteriormente ejercicios de aplicación. | |



| | |
|-----------------------------------|---------------------------------|
| Actividades de aprendizaje | Se realizan tareas y prácticas. |
|-----------------------------------|---------------------------------|

| | | |
|--|--|-------------|
| 4. Programación Visual C# | | 24 h |
| Ambiente de Desarrollo Integrado (IDE, por sus siglas en inglés) | | |
| Objetivo Específico: | Hacer programas utilizando lenguaje de programación Visual C#. | |
| <p>4.1 Interfaces Visuales (Windows Forms)</p> <p>4.2 Objetos/Elementos</p> <p>4.2.1 Controles</p> <p>4.2.1.1 Button</p> <p>4.2.1.2 TextBox</p> <p>4.2.1.3 Label</p> <p>4.2.1.4 ComboBox</p> <p>4.2.1.5 ListBox</p> <p>4.2.1.6 CheckBox</p> <p>4.2.1.7 RadioButton</p> <p>4.2.1.8 RichTextBox</p> <p>4.2.2 Contenedores</p> <p>4.2.2.1 GroupBox</p> <p>4.2.2.2 Panel</p> <p>4.2.3 Menús y barras de herramientas</p> <p>4.2.3.1 MenuStrip</p> <p>4.2.3.2 ContextMenuStrip</p> <p>4.2.4 Datos</p> <p>4.2.4.1 DataGridView</p> <p>4.2.5 Cuadros de dialogo</p> <p>4.2.5.1 OpenFileDialog</p> <p>4.2.5.2 SaveFileDialog</p> <p>4.3 Eventos</p> <p>4.4 Clases</p> <p>4.5 Excepciones</p> <p>4.5.1 Sentencias try, throw y catch</p> <p>4.5.2 Manejo de excepciones</p> <p>4.5.2.1 Recuperación de fallas</p> <p>4.6 Proyecto</p> | | |
| Lecturas y otros recursos | Se recomienda leer los temas de la bibliografía sugerida. | |
| Métodos de enseñanza | Se presentan los conceptos teóricos y posteriormente ejercicios de aplicación. | |
| Actividades de aprendizaje | Se realizan tareas y prácticas. | |

E) ESTRATEGIAS DE ENSEÑANZA Y APRENDIZAJE

Exposiciones por parte del profesor con apoyo de material audiovisual y análisis de conceptos teóricos.
Aprendizaje colaborativo.
Aprendizaje basado en problemas.

F) EVALUACIÓN Y ACREDITACIÓN

La calificación de la materia es el promedio de los tres exámenes parciales y un examen final ordinario.

| Elaboración y/o presentación de: | Periodicidad | Abarca | Ponderación |
|---|--------------|----------------------------|-------------|
| Primer parcial <ul style="list-style-type: none"> • Examen escrito teórico-práctico 70% • Actividades (tareas, actividades colaborativas, resolución de | 16 sesiones | Unidades 1, 2.1, 2.2 y 2.3 | 25 % |



| | | | |
|---|---|------------------------------|--------------|
| problemas): 30% | | | |
| Segundo parcial • Examen escrito teórico-práctico 70% • Actividades (tareas, actividades colaborativas, resolución de problemas): 30% | 32 sesiones | Unidades 2.4, 2.5, 2.6 y 2.7 | 25 % |
| Tercer parcial • Examen escrito teórico-práctico 70% • Actividades (tareas, actividades colaborativas, resolución de problemas): 30% | 48 sesiones | Unidades 3 y 4 | 25 % |
| Cuarto parcial • Proyecto 100% | 64 sesiones | Unidades 3 y 4 | 25 % |
| TOTAL ORDINARIO | | | 100% |
| Examen Extraordinario 100% Examen escrito teórico-práctico en computadora | Semana 17 del semestre en curso | Todas las unidades | 100% Temario |
| Examen a título 100% Examen escrito teórico-práctico en computadora | De acuerdo con programación de Secretaría Escolar | Todas las unidades | 100% Temario |
| Examen de regularización 100% Examen escrito teórico-práctico en computadora | De acuerdo con programación de Secretaría Escolar | Todas las unidades | 100% Temario |

G) BIBLIOGRAFÍA Y RECURSOS INFORMÁTICOS

Textos básicos:

- Cormen, Leiserson, Rivest, Stein. "Introduction to algorithms". MIT Press, 2022
- Harvey M. Deitel. "C++ Cómo Programar". Pearson, 2014
- Jhon Sharp, Microsoft Visual C# Step by Step, Pearson. 2018

Textos complementarios:

- Schildt, Herbert. "C: The Complete Reference". McGraw-Hill-Osborne 2990
- Paul Deitel, Harvey Deitel, C++ 20 for Programmers

Enlaces de Internet

- Documentación de C#
<https://learn.microsoft.com/es-es/dotnet/csharp/>
- Documentación de C++
<https://learn.microsoft.com/es-es/cpp/?view=msvc-170>
<http://cplusplus.com>

Software:

- IDE Microsoft Visual Studio
 - C++
 - C#