

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

## COORDINACIÓN GENERAL DE FORMACIÓN PROFESIONAL

### PROGRAMA DE UNIDAD DE APRENDIZAJE

#### I. DATOS DE IDENTIFICACIÓN

- 1. Unidad Académica:** Facultad de Ingeniería, Arquitectura y Diseño, Ensenada; Facultad Ciencias Químicas e Ingeniería, Tijuana; y Facultad de Ciencias de la Ingeniería y Tecnología, Valle de las Palmas
- 2. Programa Educativo:** Ingeniero en Software y Tecnologías Emergentes
- 3. Plan de Estudios:** 2022-1
- 4. Nombre de la Unidad de Aprendizaje:** Paradigmas de la Programación
- 5. Clave:** 40032
- 6. HC:** 02 **HT:** 01 **HL:** 02 **HPC:** 00 **HCL:** 00 **HE:** 02 **CR:** 07
- 7. Etapa de Formación a la que Pertenece:** Disciplinaria
- 8. Carácter de la Unidad de Aprendizaje:** Optativa
- 9. Requisitos para Cursar la Unidad de Aprendizaje:** Ninguno



#### Equipo de diseño de PUA

Claudia Margarita Rangel López  
Leticia Cervantes Huerta  
Rodrigo Lara Melgoza

#### Vo.Bo. de subdirector(es) de Unidad(es) Académica(s)

Humberto Cervantes De Ávila  
Daniela Mercedes Martínez Platas  
Noemí Hernández Hernández

**Fecha:** 20 de febrero de 2021

## **II. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE**

La finalidad de la unidad de aprendizaje es la comparación de los diferentes paradigmas de programación para el desarrollo de programas de cómputo.

La utilidad de esta radica en que le permite al estudiante determinar el paradigma de programación que debe utilizar considerando la problemática a la que se enfrenta, así como seleccionar el lenguaje de programación adecuado.

Se imparte en la etapa disciplinaria con carácter optativo. Pertenece al área de conocimiento Métodos y Tecnologías de Software.

## **III. COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE**

Desarrollar programas de cómputo, utilizando diferentes paradigmas de programación y sus lenguajes correspondientes, para comparar las características de los diferentes lenguajes de programación comerciales, con actitud creativa y sistemática.

## **IV. EVIDENCIA(S) DE APRENDIZAJE**

Repositorio de programas de cómputo (incluyendo la descripción del problema que solucionan y el algoritmo implementado) y estudio comparativo de los lenguajes de programación utilizados para contrastar ventajas y desventajas de cada lenguaje.

**V. DESARROLLO POR UNIDADES**  
**UNIDAD I. Lenguajes de programación**

**Competencia:**

Identificar la importancia de los lenguajes de programación, a partir del análisis de su historia, antecedentes y niveles de descripción, para comprender su aplicación, con actitud reflexiva, crítica e interpretativa.

**Contenido:**

**Duración:** 6 horas

- 1.1 Historia y antecedentes
- 1.2 Niveles de descripción
  - 1.2.1 Léxico y sintaxis
  - 1.2.2 Semántica
  - 1.2.3 Pragmática
  - 1.2.4 Compiladores e intérpretes

## UNIDAD II. Elementos fundamentales de los lenguajes de programación

### Competencia:

Distinguir los elementos fundamentales de los lenguajes de programación, considerando el entorno de desarrollo, su estructura y sintaxis, para utilizarlos en la creación de programas de cómputo, con actitud sistemática, creativa y con pensamiento lógico.

### Contenido:

**Duración:** 8 horas

- 2.1 Nombres y entorno
  - 2.1.1 Nombres y objetos
  - 2.1.2 Entornos y bloques
  - 2.1.3 Reglas de alcance
- 2.2 Administración de la memoria
- 2.3 Estructuras de control
  - 2.3.1 Expresiones
  - 2.3.2 Comandos
  - 2.3.3 Secuencia
  - 2.3.4 Selección
  - 2.3.5 Iteración
  - 2.3.6 Recursión
- 2.4 Subprogramas
- 2.5 Tipos de datos

### UNIDAD III. El paradigma orientado a objetos

**Competencia:**

Emplear el paradigma orientado a objetos, mediante su implementación en la resolución de problemas reales, para comprender sus ventajas y desventajas, con actitud crítica, proactiva y reflexiva.

**Contenido:**

- 3.1 Conceptos fundamentales
- 3.2 Aspectos de implementación
- 3.3 Polimorfismo y genéricos

**Duración:** 6 horas

## UNIDAD IV. El paradigma funcional

### **Competencia:**

Aplicar el paradigma funcional, mediante su implementación en la resolución de problemas reales, para comprender sus ventajas y desventajas, con actitud crítica, proactiva y reflexiva.

### **Contenido:**

- 4.1 Conceptos fundamentales
- 4.2 Evaluación
- 4.3 Programación en un lenguaje funcional

**Duración:** 6 horas

## UNIDAD V. El paradigma lógico

### **Competencia:**

Utilizar el paradigma lógico, mediante su implementación en la resolución de problemas reales, para comprender sus ventajas y desventajas, con actitud crítica, proactiva y reflexiva.

### **Contenido:**

- 5.1 Conceptos fundamentales
- 5.2 Sintaxis y el modelo computacional
- 5.3 Programación en un lenguaje lógico

**Duración:** 6 horas

## VI. ESTRUCTURA DE LAS PRÁCTICAS DE TALLER

No.	Nombre de la Práctica	Procedimiento	Recursos de Apoyo	Duración
<b>UNIDAD I</b>				
1	Lenguaje de programación	<ol style="list-style-type: none"> <li>1. Revisa los antecedentes y niveles de descripción de los lenguajes de programación.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando los niveles de descripción de los lenguajes de programación.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	3 horas
<b>UNIDAD II</b>				
2	Nombres y entorno	<ol style="list-style-type: none"> <li>1. Revisa los nombres, objetos y reglas de alcance de los lenguajes de programación.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando los nombres, objetos y reglas de alcance de los lenguajes de programación.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 Hora
3	Administración de memoria y las estructuras de control	<ol style="list-style-type: none"> <li>1. Revisa la administración de memoria y las estructuras de control de los lenguajes de programación.</li> <li>2. Analiza el problema provisto por el docente.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	2 Horas

		<ol style="list-style-type: none"> <li>3. Diseña la solución utilizando la administración de memoria adecuada y las estructuras de control de los lenguajes de programación.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>		
4	Subprogramas y tipos de datos	<ol style="list-style-type: none"> <li>1. Revisa los subprogramas y tipos de datos de los lenguajes de programación.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando los subprogramas y tipos de datos de los lenguajes de programación.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 Hora
<b>UNIDAD III</b>				
5	Conceptos fundamentales del paradigma orientado a objetos	<ol style="list-style-type: none"> <li>1. Revisa los conceptos fundamentales del paradigma orientado a objetos.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando los conceptos fundamentales de los paradigmas orientados a objetos.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora
6	Aspectos de implementación	<ol style="list-style-type: none"> <li>1. Revisa los aspectos de implementación del</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> </ul>	1 hora

		<p>paradigma orientado a objetos.</p> <ol style="list-style-type: none"> <li>Analiza el problema provisto por el docente.</li> <li>Diseña la solución utilizando los aspectos de implementación de los paradigmas orientados a objetos.</li> <li>Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>Papel o cuaderno</li> </ul>	
7	Polimorfismo y genéricos	<ol style="list-style-type: none"> <li>Revisa el polimorfismo y genéricos del paradigma orientado a objetos.</li> <li>Analiza el problema provisto por el docente.</li> <li>Diseña la solución utilizando el polimorfismo y genéricos de los paradigmas orientados a objetos.</li> <li>Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>Recursos bibliográficos</li> <li>Lápiz o pluma</li> <li>Papel o cuaderno</li> </ul>	1 hora
<b>UNIDAD IV</b>				
8	Conceptos fundamentales del paradigma funcional	<ol style="list-style-type: none"> <li>Revisa los conceptos fundamentales del paradigma funcional.</li> <li>Analiza el problema provisto por el docente.</li> <li>Diseña la solución utilizando los conceptos fundamentales del paradigma funcional.</li> <li>Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>Recursos bibliográficos</li> <li>Lápiz o pluma</li> <li>Papel o cuaderno</li> </ul>	1 hora

9	Evaluación	<ol style="list-style-type: none"> <li>1. Revisa la evaluación en el paradigma funcional.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando la evaluación del paradigma funcional.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora
10	Programación en un lenguaje funcional	<ol style="list-style-type: none"> <li>1. Revisa la programación en un lenguaje funcional.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando un lenguaje funcional.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora
<b>UNIDAD V</b>				
11	Conceptos fundamentales del paradigma lógico	<ol style="list-style-type: none"> <li>1. Revisa los conceptos fundamentales del paradigma lógico.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando los conceptos fundamentales del paradigma lógico.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora
12	Sintaxis y el modelo computacional	<ol style="list-style-type: none"> <li>1. Revisa el paradigma de lógica.</li> <li>2. Analiza el problema provisto por el docente.</li> </ol>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora

		<ol style="list-style-type: none"> <li>3. Diseña la solución utilizando el paradigma de lógica.</li> <li>4. Entrega la actividad al docente para su retroalimentación.</li> </ol>		
13	Programación en un lenguaje lógico	<ol style="list-style-type: none"> <li>1. Revisa la programación en un lenguaje lógico.</li> <li>2. Analiza el problema provisto por el docente.</li> <li>3. Diseña la solución utilizando la programación en un lenguaje lógico.</li> </ol> <p>Entrega la actividad al docente para su retroalimentación.</p>	<ul style="list-style-type: none"> <li>● Recursos bibliográficos</li> <li>● Lápiz o pluma</li> <li>● Papel o cuaderno</li> </ul>	1 hora

#### VI. ESTRUCTURA DE LAS PRÁCTICAS DE LABORATORIO

No.	Nombre de la Práctica	Procedimiento	Recursos de Apoyo	Duración
<b>UNIDAD II</b>				
1	Elementos fundamentales de los lenguajes de programación	<ol style="list-style-type: none"> <li>1. Implementar los elementos fundamentales de los lenguajes de programación               <ul style="list-style-type: none"> <li>- Nombres y entorno</li> <li>- Administración de la memoria</li> <li>- Estructuras de control</li> <li>- Subprogramas</li> <li>- Tipos de datos</li> </ul>               para la solución de un problema.             </li> <li>2. Codifica un programa que sea congruente con el</li> </ol>	<ul style="list-style-type: none"> <li>● Computadora</li> <li>● Compilador o IDE</li> </ul>	8 horas

		<p>análisis realizado para dar solución al problema.</p> <p>3. Escribe el reporte de práctica correspondiente.</p> <p>Entrega la actividad al docente para su retroalimentación.</p>		
<b>UNIDAD III</b>				
2	Paradigma orientado a objetos	<p>1. Implementa el paradigma orientado a objetos</p> <ul style="list-style-type: none"> <li>- Conceptos fundamentales</li> <li>- aspectos de implementación</li> <li>- polimorfismo y genéricos</li> </ul> <p>para dar solución a un problema dado.</p> <p>2. Codifica un programa que sea congruente con el análisis realizado para dar solución al problema.</p> <p>3. Escribe el reporte de práctica correspondiente.</p> <p>Entrega la actividad al docente para su retroalimentación.</p>	<ul style="list-style-type: none"> <li>• Computadora</li> </ul> <p>Compilador o IDE de programación orientada a objetos.</p>	8 horas
<b>UNIDAD IV</b>				
3	Paradigma funcional	<p>1. Implementa el paradigma funcional mediante sus conceptos fundamentales, evaluación y la programación en un lenguaje funcional para la solución de un problema dado.</p> <p>2. Codifica un programa que sea congruente con el</p>	<ul style="list-style-type: none"> <li>• Computadora</li> </ul> <p>Compilador o IDE de programación funcional.</p>	8 horas

		<p>análisis realizado para solucionar el problema.</p> <p>3. Escribe el reporte de práctica correspondiente.</p> <p>Entrega la actividad al docente para su retroalimentación.</p>		
<b>UNIDAD V</b>				
4	Paradigma lógico	<p>1. Implementa los conceptos fundamentales, la sintaxis y el modelo conceptual, así como la programación en un lenguaje lógico para la solución de un problema dado.</p> <p>2. Codifica un programa que sea congruente con el análisis realizado para solucionar el problema.</p> <p>3. Escribe el reporte de práctica correspondiente.</p> <p>Entrega la actividad al docente para su retroalimentación.</p>	<ul style="list-style-type: none"> <li>• Computadora</li> </ul> <p>Compilador o IDE de programación lógica.</p>	8 horas

## VII. MÉTODO DE TRABAJO

**Encuadre:** El primer día de clase el docente debe establecer la forma de trabajo, criterios de evaluación, calidad de los trabajos académicos, derechos y obligaciones docente-alumno.

**Estrategia de enseñanza (docente):**

- Estudio de caso
- Método de proyectos
- Aprendizaje basado en problemas
- Técnica expositiva
- Ejercicios prácticos

**Estrategia de aprendizaje (alumno):**

- Investigación
- Estudio de caso
- Trabajo en equipo
- Exposiciones
- Organizadores gráficos
- Cuadros comparativos
- Resolución de ejercicios en taller y laboratorio

## VIII. CRITERIOS DE EVALUACIÓN

La evaluación será llevada a cabo de forma permanente durante el desarrollo de la unidad de aprendizaje de la siguiente manera:

### **Criterios de acreditación**

- Para tener derecho a examen ordinario y extraordinario, el estudiante debe cumplir con los porcentajes de asistencia que establece el Estatuto Escolar vigente.
- Calificación en escala del 0 al 100, con un mínimo aprobatorio de 60.

### **Criterios de evaluación**

#### **Criterios de evaluación**

- Evaluaciones parciales.....	40%
- Prácticas de taller y tareas.....	30%
- Repositorio de programas de cómputo.....	30%
<b>Total.....</b>	<b>100%</b>

**Nota: la evidencia de aprendizaje debe reflejarse en este apartado y tener un porcentaje considerado en la calificación total.**

## IX. REFERENCIAS

### Básicas

Gabrielli, M, Martini, S. (2010). *Programming languages: principles and paradigms*. Alemania: Springer. [clásica].

Toal, R., Rivera, R., Schneider, A., Choe, E. (2016). *Programming language explorations*. Estados Unidos: CRC Press.

### Complementarias

Austin, T., Horstmann, C., Vue, H. (Abril, 2018). *Explicit short program practice in a programming languages course*. Journal of Computing sciences in Colleges, 33 (4), 114-122. Recuperado de <https://dl.acm.org/doi/10.5555/3199572.3199588>

Harper, R. (2017). *Practical foundations for programming languages*. Reino Unido: Cambridge University Press.

## **X. PERFIL DEL DOCENTE**

El docente que imparta la unidad de aprendizaje Paradigmas de la Programación debe contar con título de Ingeniero de software, Licenciado en ciencias de la computación o área afín, con conocimientos de lenguajes de programación para diferentes paradigmas; preferentemente con estudios de posgrado en ciencias de la computación y al menos dos años de experiencia docente. Debe ser creativo, organizado y analítico.