

# SÍLABO

## Fundamentos de Programación

<b>Código</b>	ASUC01312	<b>Carácter</b>	Obligatoria	
<b>Prerrequisito</b>	30 créditos aprobados Ninguno para la EAP de Ciencias de la Computación			
<b>Créditos</b>	4			
<b>Horas</b>	<b>Teóricas</b>	2	<b>Prácticas</b>	4
<b>Año académico</b>	2025-00			

### I. Introducción

---

Fundamentos de Programación es una asignatura obligatoria de facultad, ubicada en el primer periodo académico de la carrera profesional de Ciencias de la Computación y en el tercer periodo académico de las carreras profesionales de la Facultad de Ingeniería. Tiene como prerrequisito haber aprobado 30 créditos y ninguno para la carrera profesional de Ciencias de la Computación y es prerrequisito de las asignaturas de Estructura de Datos y Algoritmos en la carrera profesional de Ciencias de la Computación, Estructura de Datos y Programación Orientada a Objetos en la carrera profesional de Ingeniería de Sistemas e Informática y prerrequisito de la asignatura de Estructura de Datos en la carrera profesional de Ingeniería Empresarial. Con esta asignatura se desarrolla en un nivel intermedio la competencia transversal: Conocimientos de Ingeniería. La relevancia de la asignatura reside en elaborar programas para computadora, empleando un lenguaje de programación.

**Los contenidos generales que la asignatura desarrolla son:** conceptos básicos de programación: algoritmo, variable, constante, tipos de datos, operadores; estructura general de un programa; estructuras de control de programación: secuencial, selectivas y repetitivas; módulos de programa: conceptos básicos y tipos, función y procedimiento, funciones del lenguaje de programación; recursividad.

---

### II. Resultado de aprendizaje de la asignatura

---

Al finalizar la asignatura, el estudiante será capaz de implementar programas para computadora con la finalidad de resolver problemas de ingeniería.

---

**III. Organización de los aprendizajes**

<b>Unidad 1</b>		Duración en horas	<b>24</b>
<b>Estructuras de control para la programación: secuencial y selectiva</b>			
<b>Resultado de aprendizaje:</b>	Al finalizar la unidad, el estudiante será capaz de aplicar estructuras de control secuencial y selectivas, en la construcción de programas computacionales usando un lenguaje de programación.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Algoritmo: definición, características y representación</li> <li>2. Variables y tipos de datos</li> <li>3. Introducción a la programación: programas traductores, programación estructurada</li> <li>4. Estructuras de control para la programación: estructuras de control secuencial</li> <li>5. Estructuras de control para la programación: estructuras de control selectiva: simple, compuesta y múltiple</li> </ol>		
<b>Unidad 2</b>		Duración en horas	<b>24</b>
<b>Estructuras de control para la programación: repetitiva</b>			
<b>Resultado de aprendizaje:</b>	Al finalizar la unidad, el estudiante será capaz de aplicar estructuras de control repetitivas, en la construcción de programas computacionales usando un lenguaje de programación.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Estructuras de control para la programación: estructuras de control repetitivas: mientras, hacer-mientras y desde/para</li> <li>2. Creación de menú de opciones</li> <li>3. Contador y acumulador</li> </ol>		
<b>Unidad 3</b>		Duración en horas	<b>24</b>
<b>Módulos para la programación: función y procedimiento</b>			
<b>Resultado de aprendizaje:</b>	Al finalizar la unidad, el estudiante será capaz de aplicar módulos de programación y librerías de programación, en la construcción de programas computacionales.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Modularización de programas: definición y características</li> <li>2. Paso de parámetros en los módulos de programa</li> <li>3. Tipos de módulos: función y procedimiento</li> <li>4. Creación de menú de opciones con módulos de programa</li> <li>5. Librerías de programación</li> </ol>		

<b>Unidad 4</b>		Duración en horas	<b>24</b>
<b>Módulos para la programación: funciones propias del lenguaje de programación – recursividad</b>			
Resultado de aprendizaje:	Al finalizar la unidad, el estudiante será capaz de aplicar las funciones propias del lenguaje de programación, en la construcción de programas computacionales.		
Ejes temáticos:	<ol style="list-style-type: none"> <li>1. Funciones propias del lenguaje de programación: definición</li> <li>2. Recursividad</li> </ol>		

#### IV. Metodología

---

El proceso de aprendizaje se desarrolla siguiendo la secuencia teórico-práctica, planteando situaciones problemáticas de organizaciones reales, donde los estudiantes proponen alternativas de solución, ensayan sus propuestas, debaten y se consolida el saber. Se aplica a situaciones nuevas, para el efecto se utilizará:

##### **a. Modalidad Presencial - Virtual**

- Método de aprendizaje cooperativo
- Aprendizaje basado en problemas
- Aula invertida
- Aprendizaje basado en TIC
- Aprendizaje basado en retos

##### **b. Modalidad Semipresencial Virtual**

- Método de aprendizaje cooperativo
- Aprendizaje basado en problemas
- Aprendizaje basado en TIC
- Aula invertida

##### **c. Modalidad A Distancia**

- Método de aprendizaje cooperativo
  - Aprendizaje basado en problemas
  - Aprendizaje basado en TIC
-

**V. Evaluación**
**Modalidad Presencial - Virtual**

<b>Rubros</b>	<b>Unidad por evaluar</b>	<b>Fecha</b>	<b>Entregable/Instrumento</b>	<b>Peso parcial</b>	<b>Peso Total</b>
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / <b>Prueba objetiva</b>	<b>0 %</b>	
Consolidado 1 <b>C1</b>	1	Semana 1 - 4	Evaluación individual teórico-práctica / <b>Prueba mixta</b>	35 %	<b>20 %</b>
	2	Semana 5 - 7	Evaluación práctica de modo individual / <b>Prueba de desarrollo</b>	50 %	
	1 y 2	Semana 1-7	Actividades de trabajo autónomo en línea	15%	
Evaluación parcial <b>EP</b>	1 y 2	Semana 8	Exposición de los programas según los casos propuestos y será de modo grupal. / <b>Rúbrica de evaluación</b>	<b>25 %</b>	
Consolidado 2 <b>C2</b>	3	Semana 9 - 12	Programa ejecutable: Desarrollo de un programa, orientado a un problema de ingeniería, en el lenguaje C ++ aplicando la estructura modula (reto) / <b>Rúbrica de evaluación</b>	55 %	<b>20 %</b>
	4	Semana 13 - 15	Exposiciones de funciones según los casos propuestos: grupal / <b>Rúbrica de evaluación</b>	30 %	
	3 y 4	Semana 9-15	Actividades de trabajo autónomo en línea	15%	
Evaluación final <b>EF</b>	Todas las unidades	Semana 16	Exposición de los programas de cómputo con las estructuras de control, módulos de programa, librerías de programación y uso de menús de opciones, según los casos propuestos: grupal/ <b>Rúbrica de evaluación</b>	<b>35 %</b>	
Evaluación sustitutoria *	Todas las unidades	Fecha posterior a la evaluación final	<b>Aplica</b>		

\* Reemplaza la nota más baja obtenida en los rubros anteriores.

**Modalidad A Distancia**

Rubros	Unidad por evaluar	Fecha	Entregable/Instrumento	Peso
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / <b>Prueba objetiva</b>	<b>0 %</b>
Consolidado 1 <b>C1</b>	1	Semana 2	Evaluación individual teórico-práctica de modo individual/ <b>Prueba mixta</b>	<b>20 %</b>
Evaluación parcial <b>EP</b>	1 y 2	Semana 4	Evaluación práctica de modo individual/ <b>Prueba de desarrollo</b>	<b>25 %</b>
Consolidado 2 <b>C2</b>	3	Semana 6	Evaluación práctica de modo individual/ <b>Prueba de desarrollo</b>	<b>20 %</b>
Evaluación final <b>EF</b>	Todas las unidades	Semana 8	Evaluación práctica de modo individual/ <b>Prueba de desarrollo</b>	<b>35 %</b>
Evaluación sustitutoria *	Todas las unidades	Fecha posterior a la evaluación final	<b>Aplica</b>	

\* Reemplaza la nota más baja obtenida en los rubros anteriores.

**Modalidad Semipresencial -Virtual**

Rubros	Unidad por evaluar	Fecha	Entregable/Instrumento	Peso Total
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / <b>Prueba objetiva</b>	<b>0 %</b>
Consolidado 1 <b>C1</b>	1	Semana 1-3	Evaluación individual teórico-práctica / <b>Prueba mixta</b> Evaluación práctica de modo individual / <b>Prueba de desarrollo</b>	<b>20 %</b>
Evaluación parcial <b>EP</b>	1 y 2	Semana 4	Evaluación práctica de modo individual/ <b>Prueba de desarrollo</b>	<b>25 %</b>
Consolidado 2 <b>C2</b>	3	Semana 5-7	Evaluación práctica de modo individual/ <b>Prueba de desarrollo</b> Exposición de las funciones según los casos propuestos: grupal. / <b>Rúbrica de evaluación</b>	<b>20 %</b>
Evaluación final <b>EF</b>	Todas las unidades	Semana 8	Exposición de los programas de cómputo con las estructuras de control, módulos de programa, librerías de programación y uso de menús de opciones, según los casos propuestos: grupal/ <b>Rúbrica de evaluación</b>	<b>35 %</b>
Evaluación sustitutoria *	Todas las unidades	Fecha posterior a la evaluación final	<b>Aplica</b>	

\* Reemplaza la nota más baja obtenida en los rubros anteriores.

Fórmula para obtener el promedio:

$$PF = C1 (20 \%) + EP (25 \%) + C2 (20 \%) + EF (35 \%)$$

## VI. Bibliografía

### Básica

Joyanes, L. (2020). Fundamentos de programación. (5.a ed.). McGraw-Hill.  
<https://bit.ly/3YGDyyF>

### Complementaria:

Acera, M. (2017). Curso de programación C/C ++. (4° ed.) Madrid: Anaya Multimedia.  
Prialé, L. (2009). Algoritmos y diagramas de flujo aplicaciones en C++. (1° ed.) Perú:  
Megabyte.

### Recursos digitales

Dev C++. (2023). Instaladores POo y FP. Carpeta Drive.  
[https://drive.google.com/drive/folders/1QYyRtswudgddb\\_b6DwZfr5Oo4uvKyynv?usp=share\\_link](https://drive.google.com/drive/folders/1QYyRtswudgddb_b6DwZfr5Oo4uvKyynv?usp=share_link)