

SÍLABO

Fundamentos de Programación

Código	ASUC01312	Carácter	Obligatoria	
Prerrequisito	30 créditos aprobados			
Créditos	4			
Horas	Teóricas	2	Prácticas	4
Año académico	2020			

I. Introducción

Fundamentos de Programación es una asignatura obligatoria de facultad, ubicada en el tercer periodo académico de las carreras profesionales de la Facultad de Ingeniería. Tiene como prerrequisito haber aprobado 20 créditos y es prerrequisito de las asignaturas de 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

Unidad 1		Duración en horas	24
Estructuras de control para la programación: secuencial y selectiva			
Resultado de aprendizaje:	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.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Algoritmo: definición, características y representación 2. Variables y tipos de datos 3. Introducción a la programación: programas traductores, programación estructurada 4. Estructuras de control para la programación: estructuras de control secuencial 5. Estructuras de control para la programación: estructuras de control selectiva: simple, compuesta y múltiple 		

Unidad 2		Duración en horas	24
Estructuras de control para la programación: repetitiva			
Resultado de aprendizaje:	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.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Estructuras de control para la programación: estructuras de control repetitivas: mientras, hacer-mientras y desde/para 2. Creación de menú de opciones 3. Contador y acumulador 		

Unidad 3		Duración en horas	24
Módulos para la programación: función y procedimiento			
Resultado de aprendizaje:	Al finalizar la unidad, el estudiante será capaz de aplicar módulos de programación y librerías de programación, en la construcción de programas computacionales.		
Ejes temáticos:	<ol style="list-style-type: none"> 1. Modularización de programas: definición y características 2. Paso de parámetros en los módulos de programa 3. Tipos de módulos: función y procedimiento 4. Creación de menú de opciones con módulos de programa 5. Librerías de programación 		

Unidad 4		Duración en horas	24
Módulos para la programación: funciones propias del lenguaje de programación – recursividad			
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:	1. Funciones propias del lenguaje de programación: definición 2. Recursividad		

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, y se aplica a situaciones nuevas, para el efecto se utilizará:

a. Modalidad presencial:

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

b. Modalidad semipresencial - Gente que Trabaja

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

c. Modalidad semipresencial - Distancia

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

V. Evaluación
Modalidad presencial

Rubros	Unidad a evaluar	Fecha	Entregable/Instrumento	Peso Total
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / Prueba objetiva	0 %
Consolidado 1 C1	1	Semana 1-4	Evaluación individual teórico-práctica de modo individual/ Prueba mixta	20%
	2	Semana 5-7	Evaluación práctica de modo individual / Prueba de desarrollo	
Evaluación parcial EP	1 y 2	Semana 8	Exposición de los programas de cómputo con las estructuras de control secuencial, selectivas, repetitivas y uso de menú de opciones, según los casos propuestos y será de modo grupal./ Rúbrica	25%
Consolidado 2 C2	3	Semana 9-12	Evaluación práctica de modo individual/ Prueba de desarrollo	20%
	4	Semana 13-15	Exposición de las funciones propias del lenguaje de programación en los programas de cómputo, según los casos propuestos: grupal./ Rúbrica	
Evaluación final EF	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./ Rúbrica	35%
Evaluación sustitutoria*	Todas las unidades	Fecha posterior a la evaluación final	No aplica	

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

Modalidad semipresencial – Distancia

Rubros	Unidad a evaluar	Fecha	Entregable/Instrumento	Peso
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / Prueba objetiva	0 %
Consolidado 1 C1	1	Semana 2	Evaluación individual teórico-práctica de modo individual/ Prueba mixta	20 %
Evaluación parcial EP	1 y 2	Semana 4	Evaluación práctica de modo individual/ Prueba de desarrollo	25 %
Consolidado 2 C2	3	Semana 6	Evaluación práctica de modo individual/ Prueba de desarrollo	20 %
Evaluación final EF	Todas las unidades	Semana 8	Evaluación práctica de modo individual/ Prueba de desarrollo	35 %
Evaluación sustitutoria*	Todas las unidades	Fecha posterior a la evaluación final	No aplica	

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

Modalidad semipresencial (Gente que Trabaja)

Rubros	Unidad a evaluar	Fecha	Entregable/Instrumento	Peso parcial	Peso Total
Evaluación de entrada	Prerrequisito	Primera sesión	Evaluación teórico-práctica / Prueba objetiva	0 %	
Consolidado 1 C1	1	Semana 1-3	Actividad virtual	15 %	20 %
			Evaluación individual teórico-práctica de modo individual/ Prueba mixta	85 %	
Evaluación parcial EP	1 y 2	Semana 4	Evaluación práctica de modo individual/ Prueba de desarrollo	25 %	
Consolidado 2 C2	3	Semana 5-7	Actividad virtual	15 %	20 %
			Evaluación práctica de modo individual/ Prueba de desarrollo	85 %	
Evaluación final EF	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./ Rúbrica	35 %	
Evaluación sustitutoria*	Todas las unidades	Fecha posterior a la evaluación final	No aplica		

* 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. (2013). Fundamentos generales de programación. (1° ed.) México: McGraw-Hill.

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.