



Sílabo del curso Fundamentos de Programación

Marzo – julio 2024

I y II Ciclo

Ballon Alvarez, Joseph

I. Datos generales del curso

Nombre del curso:	Fundamentos de programación		
Prerrequisito:	No tiene	Código:	10306
Precedente:	Pre Cálculo	Semestre:	2024-1
Créditos:	3	Ciclo:	I (plan 2021) y II(plan 2024)
Horas semanales:	4 horas	Modalidad del curso:	Presencial
Tipo de curso y Carreras:	Obligatorio: <ul style="list-style-type: none"> Ingeniería de Tecnologías de Información y Sistemas Ingeniería Industrial y Comercial Ingeniería de Software Ingeniería en Ciencia de Datos Electivo: <ul style="list-style-type: none"> Economía y Negocios Internacionales Administración y Finanzas 	Coordinador del curso:	Joseph Ballon jballon@esan.edu.pe

II. Sumilla

El curso proporciona al estudiante los conceptos y herramientas básicas para la programación y los conceptos fundamentales de organización de una computadora. Conceptos generales. Tipos de datos. Instrucciones. Almacenamiento de datos. Lenguajes de Programación. Ciclo de vida de un proceso de desarrollo de software. Estructura general de un programa. Algoritmos y estructuras de programación. Programación modular. Arreglos. Cadenas de caracteres. Registros y archivos. Nociones de computabilidad. Considera los principios de Búsqueda y Ordenamiento.

III. Objetivo del curso

El objetivo del curso es que el estudiante aplique los conceptos fundamentales de los procesos de desarrollo de algoritmos para determinadas problemáticas y respectivamente el desarrollo del programa, lo que implicará consolidación de las etapas de análisis, diseño e implementación.

IV. Resultados de aprendizaje

Al finalizar el curso el estudiante:

- Selecciona las estructuras de programación apropiadas para dar solución a los mismos.
- Analiza y selecciona de las estructuras en función a sus características, lo que implica en entendimiento de la naturaleza de los problemas y elección de las estructuras provistas por los lenguajes de programación.
- Propone soluciones soportadas por herramientas de programación en función a requerimientos funcionales.

- La capacidad de aplicar conocimientos de matemáticas, ciencias e ingeniería en la solución de problemas complejos de ingeniería.
- La capacidad de conducir estudios de problemas de ingeniería usando conocimientos basados en la investigación y métodos de investigación incluyendo el diseño y la conducción de experimentos, el análisis y la interpretación de información, y la síntesis de información para producir conclusiones válidas.
- La capacidad de desenvolverse eficazmente como individuo, como miembro o líder de equipos diversos.
- La capacidad de identificar, formular, buscar información y analizar problemas complejos de ingeniería para llegar a conclusiones fundamentadas usando principios básicos de matemáticas, ciencias naturales y ciencias de la ingeniería.
- La capacidad de comunicarse eficazmente, mediante la comprensión y redacción de informes y documentación de diseño, la realización de exposiciones, y la transmisión y recepción de instrucciones claras.

V. Metodología

La metodología del curso en una primera etapa (15%) se tiene como característica ofrecer las clases de manera expositiva para brindar el sustento teórico al estudiante.

En la segunda etapa al final (85%) del curso se tiene al alumnado como principal fuente de participación, convirtiéndose en un método de trabajo netamente colaborativo en la búsqueda de soluciones a determinados problemas que el docente plantee, esto último en función a las diversas estructuras algorítmicas y de programación que se vayan a exponer.

En la búsqueda de consolidar el aprendizaje el docente por cada semana de trabajo dispondrá la expedición de problemas para ser resueltos en casa, por lo que sería ideal que el estudiante los resuelva para la consolidación de su aprendizaje.

Lo impartido en clases de manera analítica se consolidará con las clases impartidas en laboratorio; esto último debido a que se hará uso de una herramienta de programación que concretará lo planteado como algoritmo o pseudo código.

VI. Evaluación

El sistema de evaluación es continuo e integral. Comprende la nota de evaluación permanente (60%) que comprende prácticas calificadas, proyecto integrados y participación en clases, la del examen parcial (20%) y la del examen final (20%).

Las ponderaciones al interior de la evaluación permanente se describen en el cuadro siguiente.

PROMEDIO DE EVALUACIÓN PERMANENTE (PEP) 60%		
Tipo de evaluación	Descripción	Ponderación %
Prácticas Calificadas	Cuatro prácticas calificadas (Se elimina la menor nota)	50

	En la 3ra práctica se contemplará, como evaluación teórica una de las conferencias del EmprendESAN	
Proyecto integrador	Primer avance	5
	Presentación final	15
Participación en clases	Talleres	25
	Participación en aula	5

El promedio final (PF) se obtiene del siguiente modo:

$$PF = (0,20 \times EP) + (0,60 \times PEP) + (0,20 \times EF)$$

Donde:

PF = Promedio Final

EP = Examen Parcial

PEP = Promedio de evaluación permanente y

EF = Examen Final

VII. Contenido programado del curso

SEMANA	CONTENIDOS	ACTIVIDADES / EVALUACIÓN
UNIDAD DE APRENDIZAJE I: INTRODUCCIÓN A LA PROGRAMACIÓN RESULTADO DE APRENDIZAJE: <ul style="list-style-type: none"> • Selecciona las estructuras de programación apropiadas para dar solución a los mismos. • La capacidad de aplicar conocimientos de matemáticas, ciencias e ingeniería en la solución de problemas complejos de ingeniería. • La capacidad de conducir estudios de problemas de ingeniería usando conocimientos basados en la investigación y métodos de investigación incluyendo el diseño y la conducción de experimentos, el análisis y la interpretación de información, y la síntesis de información para producir conclusiones válidas. • La capacidad de desenvolverse eficazmente como individuo, como miembro o líder de equipos diversos. 		
1° Del 21 al 27 de marzo	INTRODUCCIÓN A LA PROGRAMACIÓN 1.1. Conceptos Introdutorios 1.2. Introducción a la algorítmica. Descripción de software y lenguajes de programación. 1.3. Fases para la resolución de un algoritmo. 1.4. Herramientas y tipos de programación. 1.5. Tipos de datos, Constantes, variables, identificadores, expresiones, funciones. Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 2 (Metodología de la programación y desarrollo de software) Págs. 45-53.	Presentación de la Metodología del curso Revisión de guía (pautas) para el desarrollo del trabajo final de investigación
2° Del 01 al 06 de abril	ESTRUCTURAS SECUENCIALES 2.1. Estructura de algoritmos y/o pseudocódigo. 2.2. Estructura de un Programa. 2.3. Contadores, Acumuladores. Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 2 (Metodología de la programación y desarrollo de software) Págs. 64-71.	Ejercicios Esquemas de programación
3° Del 08 al 13 de abril	PRÁCTICA CALIFICADA I 3.1 Repaso en clases 3.2 Práctica calificada	1ra. Práctica calificada (Desde 1.1 hasta el 2.3 Contadores, Acumuladores).

4° Del 15 al 20 de abril	ESTRUCTURAS SELECTIVAS I 4.1 Estructuras de Control. Programación Estructurada. 4.2 Estructuras de Control – Tipos. Estructura Selectiva simple (SI).	Ejercicios resueltos
	Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 4 (Flujo de control I: Estructuras selectivas) Págs. 127-131.	
5° Del 22 al 27 de abril	ESTRUCTURAS SELECTIVAS II 5.1 Estructura selectiva doble (SI - SINO). 5.2 Selectiva múltiple (SI - SINO - SI). 5.3 Estructura selectiva de control (SEGÚN -SEA)	Ejercicios Estructuras selectivas e iterativas
	Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 4 (Flujo de control I: Estructuras selectivas) Págs. 132-144.	
6° Del 29 de abril al 04 de mayo	PRÁCTICA CALIFICADA II 6.1 Repaso en clases 6.2 Práctica calificada	2da. Práctica calificada (Hasta el 5.3 estructura selectiva de control).
UNIDAD DE APRENDIZAJE II: MODULARIDAD EN LA PROGRAMACIÓN RESULTADO DE APRENDIZAJE: <ul style="list-style-type: none"> • Analiza y selecciona de las estructuras en función a sus características, lo que implica en entendimiento de la naturaleza de los problemas y elección de las estructuras provistas por los lenguajes de programación. • La capacidad de identificar, formular, buscar información y analizar problemas complejos de ingeniería para llegar a conclusiones fundamentadas usando principios básicos de matemáticas, ciencias naturales y ciencias de la ingeniería. 		
7° Del 06 al 11 de mayo	ESTRUCTURAS ITERATIVAS I 7.1 Estructura Iterativa (REPETIR - FOR). 7.2 Estructura Iterativa (REPETIR – MIENTRAS).	Ejercicios resueltos
	Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 5 (Flujo de control II: Estructuras repetitivas) Págs. 157-165.	
8° Del 13 al 18 de mayo	EXÁMENES PARCIALES	
9° Del 20 al 25 de mayo	ARREGLOS I 9.1 Clases estructuradas. 9.2 Tipos. Arreglos unidimensionales, declaración y acceso.	Ejercicios resueltos
	Joyanes. Fundamentos de Programación. Algoritmos,	

	estructura de datos y objetos. 4ta. Ed. Cap. 7 (Estructura de datos I (arrays y estructuras)) Págs. 247-255.	
10° Del 27 de mayo al 01 de junio	ARREGLOS II 10.1 Arreglos bidimensionales, declaración y acceso. 10.2 Ejercicios de arreglos bidimensionales Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 7 (Estructura de datos I (arrays y estructuras)) Págs. 258-260.	
11° Del 03 al 08 de junio	11.1 Repaso en clases 11.2 Práctica calificada	6 de Junio EmprendESAN (Asistencia obligatoria alumnos de Ingeniería)
UNIDAD DE APRENDIZAJE III: PROGRAMACIÓN MODULAR Y PROYECTO RESULTADO DE APRENDIZAJE: <ul style="list-style-type: none"> • Propondrá soluciones soportadas por herramientas de programación en función a requerimientos funcionales. • La capacidad de comunicarse eficazmente, mediante la comprensión y redacción de informes y documentación de diseño, la realización de exposiciones, y la transmisión y recepción de instrucciones claras. 		
12° Del 10 al 15 de junio	PROGRAMACIÓN MODULAR I 12.1 Programación Modular. Subprogramas – Tipos. 12.2 Variables globales y locales. Funciones, paso de parámetros. Métodos de paso de parámetros. Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 6 (Subprogramas (subalgoritmos): Funciones) Págs. 201-225.	3ra. Práctica calificada (Hasta el 10.1 arreglos bidimensionales).
13° Del 17 al 22 de junio	PROGRAMACIÓN MODULAR II 13.1 Creación librerías 13.2 Diseño Top-down para desarrollo de software 13.3 Idea de proyecto en top-down Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 6 (Subprogramas (subalgoritmos): Funciones) Págs. 201-225.	Ejercicios resueltos
14° Del 24 al 28 de junio	MANEJO DE ARCHIVOS Y CADENAS 14.1 Manejo de Archivos. 14.2 Funciones de Registro y Lectura 14.3 Desarrollo de proyecto. Joyanes. Fundamentos de Programación. Algoritmos, estructura de datos y objetos. 4ta. Ed. Cap. 9 (Archivos (ficheros)) Págs. 307-324.	Ejercicios resueltos

15° Del 01 al 06 de julio	Sustentación de proyectos	4ta. Práctica calificada (Desde 12.1 hasta el 14.2).
16° Del 08 al 13 de julio	EXÁMENES FINALES	

VIII. Referencias

Bibliografía Básica:

- Joyanes, L. (2020). Fundamentos de programación. (5.a ed.). McGraw-Hill.

Bibliografía Complementaria:

- Arias E. R. (2015). *Fundamentos de Programacion: Para Todos los Públicos*.
- Bruce Eckel. (2007): *Piensa en Java*. Mexico: Pearson Prentice-Hall.
- Deitel H.M. & Deitel P.J. (2004). *Como programar en C++ y Java*. Mexico: Pearson Prentice-Hall.
- Pilgrim, M. (2009). *Dive into Python 3*. New York: Apress.
- James Goslin (2014). *The java Programming Language*. EUA: Oracle Press.
- Joyanes Aguilar, Luis. (2007). *Programación orientada a objetos*. España: Mc Graw Hill.
- Smith J. (2015). *Fundamentos de programación aplicados a C++: Manual imprescindible para todos los usuarios que se inician en la programación de aplicaciones informáticas*. CreateSpace.
- Vozmediano A. M. (2017). *Aprender a programar en C: de 0 a 99 en un solo libro: Un viaje desde la programación estructurada en pseudocódigo hasta las estructuras de datos avanzadas en lenguaje C*.
- Downey, A. (2015). *Think Python*. Sebastopol, CA: O'Reilly Media.

IX. Soporte de laboratorio

Se hará uso del laboratorio de cómputo para clases prácticas y Laboratorios calificados.

Se utilizará como software el lenguaje de programación: C++ ó Python

Software a utilizar:

- IDE Code Blocks /Visual Studio 2017
- PyCharm

X. Profesores

Ballon Alvarez, Joseph
jballon@esan.edu.pe

Monzón Rojas, Elmo
emonzon@esan.edu.pe