

SILABO

FACULTAD
ESCUELA PROFESIONAL
PROGRAMA DE ESTUDIOS

INGENIERIA ESTADISTICA E INFORMATICA
INGENIERIA ESTADISTICA E INFORMATICA
CARRERA PURA

I. INFORMACIÓN GENERAL

1.1 Identificación Académica

a)	Curso	LENGUAJES DE PROGRAMACION I
b)	Código	EST303
c)	Prerequisito	EST301 - ANALISIS Y DISEÑOS DE ALGORITMOS
d)	Número de Horas	03h teóricas, 02h prácticas, Total 05 horas
e)	Créditos	04
f)	Número de Horas virtuales	04
g)	Año y Semestre Académico	2025-II
h)	Ciclo de Estudios	III
i)	Duración	Del 25 de Agosto al 19 de Diciembre del 2025 (17 semanas)
j)	Área Curricular	Estudios de especialidad
k)	Características del Curso	I +D+I, Experiencia preprofesional

1.2 Docente

a)	Apellidos y Nombres	JUAREZ VARGAS JUAN CARLOS
b)	Condición y Categoría	AUXILITAR T.C
c)	Especialidad	ING. ESTADISTICO E INFORMATICO
a)	Apellidos y Nombres	TUMI FIGUEROA ERNESTO NAYER
b)	Condición y Categoría	PRINCIPAL T.C.
c)	Especialidad	ING. ESTADISTICO

1.3 Ambiente donde se realizó el aprendizaje

- a) Aula 201 y 202 – (Pabellón antiguo FINESI) y Laboratorios 01 y 02 del pabellón de la Facultad de Ingeniería Estadística e Informática y plataforma virtual

II. SUMILLA

El curso de Lenguaje de Programación I corresponde al área curricular de especialidad, es de naturaleza teórico-práctico, cuyo propósito es desarrollar una comprensión integral del paradigma estructurado, permitiendo a los estudiantes aplicar conceptos fundamentales y técnicas prácticas para resolver problemas y construir software básico, evaluando las ventajas y desventajas de un lenguaje de programación con actitud científica y tecnológica. El curso está dividido en:

- I. Unidad I : Representación de datos, información y estructuras de control
- II. Unidad II: Archivos e Introducción a la Programación Orientada a Objetos

III. PERFIL DEL EGRESADO EN RELACIÓN AL CURSO

CE1. Diseña y desarrolla soluciones informáticas, aplicaciones web y soluciones empresariales bajo estándares internacionales para aportar con el desarrollo tecnológico, asumiendo una actitud científica y responsable utilizando principios, técnicas, herramientas y métodos que aseguren la funcionalidad y calidad del software.

IV. LOGRO DE APRENDIZAJE DEL CURSO

Explica los conceptos y principios de un lenguaje de programación y construye programas básicos por computadora

V. TRATAMIENTO DE UNIDADES DIDÁCTICAS

UNIDAD 1		UNIDAD 1
LOGROS DE APRENDIZAJE DE LA UNIDAD Comprende adecuadamente los conceptos y desarrollo aplicaciones básicas con base a los principios de programación.		
TIEMPO DE DESARROLLO		Del 25 de Agosto al 27 de Octubre del 2025 (Total 45 horas)
HORAS DE ENSEÑANZA VIRTUAL/UNIDAD		02
SEMANAS	CRITERIOS DE DESEMPEÑO	CONOCIMIENTOS
Semana 1	Comprende y explica los fundamentos de los lenguajes de programación con reflexión.	La programación y el procesamiento de datos, uso y gestión de memoria del computador, Como se ejecuta un programa en el hardware, concepto de abstracción, algoritmos
Semana 2	Discrimina adecuadamente los tipos y lenguajes de programación apropiadamente.	Tipos de lenguajes (imperativos, funcionales, lógicos, orientado a objetos, híbridos,...), lenguajes compilados, compilación y enlace de un programa, lenguajes pre compilados y máquinas virtuales, lenguajes interpretados y script, interpretación y ejecución de código en el computador, programación a bajo nivel, intermedio y alto nivel

Semana 3	Comprende y explica el pensamiento computacional aplicado a la realidad	Análisis y descomposición de un problema, abstracción de información relevante a un problema, reconocimiento de patrones repetitivos, pensamiento lógico, razonamiento algorítmico, depuración y validación de soluciones.
Semana 4	Describe los elementos de un programa de computadora en general viable.	Módulo, inicio y fin de un programa, instrucciones, datos, variables y constantes, palabras reservadas, sentencias, asignaciones y operaciones, operadores matemáticos, lógicos y relacionales, entrada y salida de datos. - Evidencia1: Planteamiento de solución de un problema con el pensamiento computacional. - Retroalimentación sobre pensamiento computacional y fundamentos de los lenguajes de programación.
Semana 5	Comprende y aplica la programación estructurada: Secuencia y condición simple con propiedad	Lógica secuencial de un programa, condición simple (if – then, condición múltiple (if – then – else, if – then – else anidado), condición múltiple (switch – case)
Semana 6	Comprende y aplica la programación estructurada: bucles de repetición “mientras” adecuado al problema	Bucles de repetición (while), bucles de repetición (do – while), romper y saltar un bucle (Break y continue)
Semana 7	Manejo de estructuras de datos básicas de los lenguajes: “arreglos” y bucles “para” con propiedad.	Estructuras estáticas básicas (arrays), manipulación de datos homogéneos en arreglos con bucles (for).
Semana 8	Implementa estructuras de repetición anidadas correctamente	Bucles anidados, bucles dobles y de más dimensiones. Manejo de matrices y arreglos multidimensionales.
Semana 9	Evaluación parcial teórica y práctica de la unidad	Evidencia 2, parciales: - Examen de resolución de problemas - Presentación y defensa de trabajos asignados - Retroalimentación pos evaluación parcial
PORCENTAJE DE AVANCE ACADÉMICO DE LA UNIDAD: 50%		

* El docente deberá programar “actividades de retroalimentación” los cuales se desarrollarán a lo largo de cada unidad didáctica con el fin de asegurar los adecuados aprendizajes de los estudiantes.

UNIDAD 2		UNIDAD 2
LOGROS DE APRENDIZAJE DE LA UNIDAD		
Construye programas básicos por computadora haciendo uso de módulos y manejo de datos del Lenguaje de Programación		
TIEMPO DE DESARROLLO		Del 27 de Octubre al 19 de Diciembre del 2025 (Total 40 horas)
HORAS DE ENSEÑANZA VIRTUAL/UNIDAD		02
SEMANAS	CRITERIOS DE DESEMPEÑO	CONOCIMIENTOS
Semana 10	Implementa de funciones y librerías definidas por el usuario con propiedad	Funciones definidas por el usuario, paso de parámetros por valor, por referencia y por dirección, creación de librerías de funciones.
Semana 11	Implementa algoritmos para generar archivos de texto y binario que permitan almacenar información estructurada	Estructuras Uniones y almacenamiento en Archivos, funciones básicas para manejo de archivos, tipos de archivos de texto y binarios
Semana 12	Construye punteros y administra la memoria en forma eficiente	Puntero, direccionamiento de memoria, Memoria estática vs. Dinámica, Operadores new y delete, Arrays dinámicos Punteros a punteros (doble indirección),Punteros constantes vs. punteros a constantes
Semana 13	Utiliza punteros a funciones, arreglo de funciones y optimizar el uso de memoria	Puntero a funciones, Arrays de punteros a funciones, Paso de funciones como parámetros, Typedef con punteros a funciones. Evidencia 3: - Informe - Defensa
Semana 14	Usa la programación orientada a objetos para mejorar la modularidad de la programación	Programación Orientada a Objetos, Análisis orientados a objetos, encapsulación, Principios de debugging y testing,
Semana 15	Desarrolla utilizando paradigmas de programación orientados a objetos	Visibilidad de atributos, métodos y constructores Herencia, Herencia Múltiple,
Semana 16	Utiliza características de programación orientada a objetos para mejorar el proceso de desarrollo.	Clases abstractas, Polimorfismo, vinculación dinámica, Interfaces, Gestión de excepciones, Enumerados, Arrays de Objetos y Plantillas de Clases
Semana 17	Evaluación parcial teórica y práctica de la unidad	Evidencia 4, parciales: - Examen de resolución de problemas - Presentación y defensa de trabajos asignados

PORCENTAJE DE AVANCE ACADÉMICO DE LA UNIDAD: 50%

* El docente deberá programar “actividades de retroalimentación” los cuales se desarrollarán a lo largo de cada unidad didáctica con el fin de asegurar los adecuados aprendizajes de los estudiantes.

VI. ESTRATEGIAS METODOLÓGICAS**6.1 De Enseñanza**

- Actividades síncronas: o Actividades en video conferencias. o Implementación de foros de discusión en línea o Uso de sala de chat en línea.
- Actividades asíncronas: o Revisión de foros de información sobre la temática o Planteamiento y revisión de tareas o Uso de materiales de lectura o Proporcionar diapositivas con la temática o Disponibilidad de vídeos o Estudio a través de Podcats

6.2 De Aprendizaje

- Análisis de textos y artículos
- Revisión de guías y manuales
- Debate, análisis y discriminación de ideas.
- Uso de software de lenguajes de programación y plataformas de programación en Pc y la nube. - Resolución de casos y exposición de propuestas de solución y análisis de soluciones

6.3 De Investigación Formativa

- Revisión de Artículos Científicos referentes a los lenguajes de programación
- Presentación de informes y trabajos usando las normas y herramientas de publicación de artículos y reportes de investigación (estructura, partes, gráficos, citas, referencias, similaridad).

6.4 De Responsabilidad Social Universitaria

- Recoger problemas de la sociedad a ser abordados y proponer soluciones buscando optimizar sus resultados y socializarlos en la realidad y difundir a través de la web.
- visita a organizaciones para proponer alternativas tecnológicas de uso

6.5 De Enseñanza Virtual

- Uso de aula virtual, materiales digitales y recursos de aprendizaje de la nube así como la administración y almacenamiento del trabajo en la nube

VII. MEDIOS Y MATERIALES DIDÁCTICOS

- Uso de plataformas web, máquinas virtuales y servicios web. - Diapositivas, manuales y guías de aprendizaje digital - Materiales en vídeo y pizarra en línea.
- Recursos de gamificación en la web.
- Software IDE y compiladores para Lenguajes de Programación (C/C++)
- Lenguaje de programación C++
- Plataformas virtuales de programación para C/C++

VIII. EVALUACIÓN DEL APRENDIZAJE**8.1 Logro de aprendizaje, evidencias de desempeño, ponderación, técnicas e instrumentos de evaluación.**

UNIDAD	LOGROS DE APRENDIZAJE	EVIDENCIAS DESEMPEÑO: De acción, objeto o producto (%)	PONDERACIÓN (Obligatorio en base a 100%)	TÉCNICAS	INSTRUMENTOS
1	Comprende adecuadamente los conceptos y desarrollo aplicaciones básicas con base a los principios de programación.	Evidencia 1: - Informe - Defensa Evidencia 2: - Examen t/p - Resolución de casos - Implementación	50%	- Revisión de informe - Observación - Examen - Análisis de resultados - Observación	- Lista de cotejo - Registro - Rúbrica - Lista de cotejo - Informe
2	Construye programas básicos por computadora haciendo uso de módulos y manejo de datos del Lenguaje de Programación	Evidencia 3: - Informe - Defensa Evidencia 4: Proyecto de Programación	50%	- Revisión de informe - Observación - Examen - Análisis de resultados - Observación	- Lista de cotejo - Registro - Rúbrica - Lista de cotejo - Informe

8.2 Evidencias de aprendizaje del semestre académico.

LOGRO DE APRENDIZAJE DEL CURSO	EVIDENCIAS: ACCIÓN, OBJETO o PRODUCTO	FECHA DE PRESENTACIÓN
Explica los conceptos y principios de un lenguaje de programación y construye programas básicos por computadora	Producto de software implementado (código y documentación) y el informe final de presentación y defensa del producto	Del 31 de agosto al 29 de diciembre del 2025.

- Evidencias que deben ser socializados con la comunidad como parte de la responsabilidad social universitaria. Por la emergencia sanitaria podría optarse por la difusión vía página web y redes sociales.

8.3 Calificación

La fórmula para la obtención del promedio final del curso es la siguiente:

$$\text{Promedio Final} = (50\%)IUPP + (50\%)IIUPP$$

Donde:

IUPP : Primero unidad promedio parcial

IIUPP : Segundo unidad promedio parcial

$$\text{Promedio parcial de la unidad} = \frac{30\% (LA1) + 70\% (LA2)}{100\%}$$

LA1 = Logro de aprendizaje 1.

LA2 = Logro de aprendizaje 2.

La fórmula para la obtención del promedio final del curso es la siguiente:

$$\text{Promedio final} = \frac{I \text{ UPP} + II \text{ UPP}}{2}$$

Hay puntos extra por mejor trabajos presentados y suma de participaciones en clases, problemas de retos en programación y otras habilidades competitivas

IX. FUENTES DE INFORMACIÓN

9.1 Bibliográficas

Básica

- SAVITCH, W., (2015). Problem Solving with C++ 9th Ed. Pearson Education. ISBN 10: 0-13-359174-3.
- FAGER, J., PANTOJA, L., & Otros (2014). Estructura de Datos. Editorial Proyecto LATIn.
- STROUSTRUP, B. (2013). The C++ programming language. 4th ed. Upper Saddle River, NJ: Addison-Wesley3. ISBN 9780321563842.
- NYHOFF. (2005). TADs, estructuras de datos y resolución de problemas con C++. Pearson Prentice Hall, 2005
- AHO, HOPCROFT, ULLMAN. (1998) Estructuras de datos y algoritmos. Addison-Wesley Longman,
- WEISS, M.A. (2014). Data structures and algorithm analysis in C++. 4th ed int. Boston: Pearson. ISBN 0273769383.
- NEAPOLITAN, R.E. (2015). Foundations of algorithms. 5th ed. Burlington, MA.: Jones and Bartlett Learning, 2015. ISBN 9781284049190
- NARASIMHA, K., (2017). Data Structures And Algorithms Made Easy. 5th Ed. CareerMonk Publications.
- SAHNI, SARTAJ. (2005). Data Structures, Algorithms And Applications in C++. 2nd. Ed. Universities Press.
- DROZDEK, ADAM. (2013). Data Structures and Algorithms in C++. 4th. Ed. CENGAGE Learning.
- MALIK, D., (2010). Data Structures Using C++. 2nd. Ed. Course Technology. CENGAGE Learning.

Complementarias

- HAREL, D.; FELDMAN, Y. (2012). Algorithmics: the spirit of computing. 3rd ed. Berlin: Springer ISBN 9783642441356.
- DASGUPTA, S.; PAPADIMITRIOU, C.; VAZIRANI, U. (2008). Algorithms. Boston. Mc Graw Hill Higher Education, 2008. ISBN - 9780073523408.
- SEDGEWICK, R. (2002). Algorithms in C++. 3rd ed. Boston: Addison-Wesley,. ISBN 0201350882.
- CORMEN, T.H.; LEISERSON, C.E.; RIVEST, R.L.; STEIN, C. (2009). Introduction to algorithms. 3rd ed. Cambridge: MIT Press.
- ROBERGÉ, J., BRANDLE, S., WHITTINGTON, D., (2003). C++ Data Structures. 2th. Ed. Jones and Barlett Publisher, Inc.
- SHAFFER, C., (2010). A Practical Introduction to Data Structures and Algorithm Analysis, 3th. Ed. Dpto. CS Virginia Tech.

Electrónicas

- Curso de programación en C++ Apuntes de clase <https://www.uv.es>
- Aprende sobre programación C + + con cursos online edX: <https://www.edx.org>
- Curso para programar C++; fácil, rápido y paso a paso: <https://www.programarya.com>
- El lenguaje C++ — Fundamentos de Programación en C++: <https://www2.eii.uva.es>
- Programación C/C++ - Con Clase: <https://conclase.net>
- C++: Qué es, Para qué sirve, Ventajas y Desventajas – Glosario: <https://i.workana.com>
- C++ | Aprende Programación Competitiva: <https://aprende.olimpiada-informatica.org>
- https://files.gitter.im/SamZhangQingChuan/sam/DA1g/Steven-Halim_-Felix-Halim-Competitive-Programming-3_-The-New-Lower-Bound-of-Programming-Contests-Lulu.com-_2013_.pdf
- <https://cses.fi/book/book.pdf>

Producción intelectual del docente relacionado con el curso

- Manuales, Diapositivas o Video tutoriales entregados al estudiante

Puno, Diciembre del 2025