



Programa de estudios por competencias
Ingeniería en Mecánica Eléctrica

1. IDENTIFICACIÓN DEL CURSO

Centro Universitario:

CENTRO UNIVERSITARIO DEL NORTE

Departamento:

FUNDAMENTOS DEL CONOCIMIENTO

Academia:

CIENCIAS COMPUTACIONALES

Nombre de la unidad aprendizaje:

PROGRAMACIÓN APLICADA

Clave de la materia:	Horas de Teoría:	Horas de practica:	Total de Horas:	Valor en créditos:
17404	34	17	51	6

Tipo de Curso:	Nivel en que se ubica:	Carrera:	Prerrequisitos:
CURSO TALLER	PREGRADO	MECANICA ELECTRICA	SIN PRERREQUISITOS

Área de formación

Básica común

Actualizado por:

Firma

ALFONSO AVALOS LEDESMA

Fecha de última actualización:

31 de octubre del 2023

2. PRESENTACIÓN

La programación es esencial en el desarrollo de software y en la resolución de problemas, permitiendo optimizar procesos, automatizar tareas y encontrar soluciones eficientes. Dominar esta disciplina es fundamental para cualquier programador o profesional donde se busque mejorar sus habilidades e innovar, así como maximizar su productividad. Existen diferentes lenguajes de programación que permiten implementar algoritmos de manera efectiva, como

Python, C++, Java, entre otros. Estos lenguajes proporcionan herramientas y estructuras de datos que facilitan la implementación de algoritmos y su posterior ejecución en un entorno de programación.

3. UNIDAD DE COMPETENCIA

Al finalizar un curso de programación aplicada, el alumno deberá tener conocimientos sólidos sobre los conceptos fundamentales de la programación, habilidades en el uso del lenguaje de programación c++ y pseint, conocimientos básicos de programación, generación de algoritmos y diagramas de flujo y habilidades de resolución de problemas a base de código. Estos conocimientos y saberes sentarán las bases para su desarrollo y crecimiento en el campo de la programación.

4. SABERES

Saberes teóricos	<ul style="list-style-type: none">● Conoce las bases básicas para la generación de algoritmos, diagramas de flujo, pseudocódigo y código de programación.● Identifica la solución del problema mediante algoritmos y diagramas de flujo.● Adquiere bases de la percepción lógica ayudando en la realización de otras actividades.
Saberes prácticos	<ul style="list-style-type: none">● Gestiona información de diferentes fuentes para la evaluación.● Realiza la evaluación de cada una de las actividades del curso.● Genera código para la resolución de problemas.
Saberes Formativos (actitudes y valores)	<ul style="list-style-type: none">● Colabora activamente en actividades del curso.● Trabaja en equipo de manera cooperativa y colaborativa.

5. CONTENIDOS

Unidad de competencia: I. Nociones básicas de programación

- 1.1. ¿Qué es la programación?
- 1.2. Definición de programación y funciones prácticas.
- 1.3. Fases para la creación de un programa.

Unidad de competencia II. Desarrollo de Algoritmos

- 2.1. Definición
- 2.2. Pensamiento algorítmico
- 2.3. Tipos de algoritmos
- 2.4. Algoritmos de búsqueda
- 2.5. Algoritmos de ordenamiento
- 2.6. Algoritmos de voraces
- 2.7. Aplicaciones prácticas

Unidad de competencia III. Pseudocódigo y diagrama de flujo

- 3.1. Definición
- 3.2. Símbolos y convenciones
- 3.3. Estructuras básicas
- 3.4. Diseño y organización
- 3.5. Aplicaciones prácticas

Unidad de competencia IV. Programación organizada y secuencias de control

- 4.1. Tipos de datos
- 4.2. Variables
- 4.3. Declaración de variables
- 4.4. Constantes
- 4.5. Operadores
- 4.6. Tipos de datos complejos
- 4.7. Expresiones
- 4.8. Primer uso del lenguaje C++
- 4.9.** Palabras reservadas

Unidad de competencia V. Estructuras de control

- 5.1. Definición
- 5.2. Estructuras de control
 - If-else if
 - For
 - While
 - Do-While
 - Switch
- 5.3. Repetitividad
- 5.4. Funciones

6. ACTIVIDADES PRÁCTICAS

Programa Pseint

- Conociendo Pseint.
- Diseño del primer algoritmo.
- Funciones “Leer y Escribir”.
- Función “Asignar”.
- Función “Si-Entonces”.
- Función “Según”.
- Función “Mientras”.
- Función “Repetir”.
- Función “Para”.

Programa C++

- Conociendo C++.
- Mi primer programa.
- Variables-Constantes
- Ciclo IF-else if
- Ciclo FOR
- Ciclo While
- Ciclo Do - While
- Ciclo Switch – Case
- Funciones.

7. METODOLOGÍA

La metodología usada para este curso es la Bada en proyectos en donde los docentes asesoran y orientan, motivan, proporcionan herramientas, recursos, así como un marco de referencia teórico que dé una guía y permita la complementariedad entre las sesiones de clase y las sesiones de práctica en la formulación y creación de algoritmos, diagramas de flujo y código.

Para la programación de algoritmos involucra comprender el problema, planificar, diseñar, implementar, probar y documentar el algoritmo. Siguiendo estos pasos, se puede desarrollar algoritmos eficientes y efectivos que resuelvan problemas de manera exitosa desafiando al alumnado a investigar, aprender y aplicar lo aprendido en un caso real, compartiendo su experiencia y analizando los resultados.

8. PROCESO DE ENSEÑANZA APRENDIZAJE

- Proceso de investigación, búsqueda, tratamiento y análisis de la información.
- Interacciones entre estudiantes y docentes (orientación, seguimiento).
- Práctica de los aprendido, los alumnos deberán de seguir generando código para poder entender cada una de las posibles soluciones del problema.
- Creación de un producto final que los alumnos eligen (una problemática real para ser optimizada o solucionada mediante la creación de código.)

9. PERFIL DEL PROFESOR

Estudios de licenciatura y posgrado dentro de las siguientes áreas: sistemas digitales, sistemas computacionales, informática, ingeniería en computación.
Debe tener un sólido conocimiento en programación, experiencia en el uso de lenguajes de programación, habilidades de enseñanza efectivas, así como creatividad y capacidad de resolución de problemas.

10. EVALUACIÓN

Actividades preliminares	10 %
Actividades de aprendizaje	20 %
Actividades integradoras	20%
Evaluaciones	20 %
Producto integrador global	30 %
TOTAL	100 %

11. BIBLIOGRAFÍA

BIBLIOGRAFÍA BÁSICA

Chavez Torres A. (2017). *Aprenda a diseñar algoritmos*. Sello Editorial UNAD.

Mancilla Herrera, A. Ebratt Gómez, R. Capacho Portilla, J. (2016). Universidad del Norte.

Ruíz Enríquez H. (2014). *Fundamentos de programación usando Pseint*. Universidad Politécnica del Carchi. ISBN: 978-9942-914-18-7. En <https://www.publicacionesupec.org/index.php/carchi/catalog/view/64/62/193>

Sznajdleder, P. A. (2021). *Curso de algoritmos y programación a fondo: implementaciones en C++*. Alfaomega.

Villegas Jaramillo E. y Guerrero Mendieta L. E. (2016). *Análisis y diseño de algoritmos: Un enfoque práctico*. Universidad Nacional de Colombia.

BIBLIOGRAFÍA COMPLEMENTARIA

Corona Nakamura, M. A. (2011). *Diseño de algoritmos y su codificación en lenguaje C*. MacGrawhill.

Méndez Giron, A. M. (2015). *Diseño de algoritmos y su programación*. Alfaomega.

Vivar, J. M. F. (2018). *Algoritmos, aplicaciones y Big Data, nuevos paradigmas en el proceso de comunicación y de enseñanza-aprendizaje del periodismo de datos*. *Revista de comunicación*, 17(2), 268-291. <https://doi.org/10.26441/rc17.2-2018-a12>

Vo.Bo

Dra. Teresa de Jesús Cárdenas Gándara
PRESIDENTE DE ACADEMIA



CENTRO UNIVERSITARIO DEL NORTE
Departamento de Fundamentos del Conocimiento

Vo.Bo

Mtra. Silvia Elena Mota Macías
PRESIDENTE DEL COLEGIO DEPARTAMENTAL