



ÁREA CURRICULAR: FORMATIVA O PROFESIONAL
SÍLABO
SOFTWARE DE PROGRAMACIÓN Y SIMULACIÓN

I. INFORMACIÓN GENERAL

1.1	Departamento Académico	:	Ingeniería Eléctrica
1.2	Semestre Académico	:	2022-A
1.3	Código de la asignatura	:	EE201
1.4	Ciclo	:	II
1.5	Créditos	:	3
1.6	Horas lectivas (Teoría, Práctica)	:	4(T=2, P=2)
1.7	Condición del curso	:	Obligatorio
1.8	Requisito(s)	:	EG103 Computación Aplicado a la Ingeniería.
1.9	Docente	:	Msc. Ing. Apesteigua Infantes Juan Antonio

II. SUMILLA

El curso pertenece al área de estudios específicos, es de naturaleza teórica y experimental, tiene el propósito de brindar a los estudiantes los elementos necesarios para desarrollar algoritmos para la resolución de aplicaciones científicas y matemáticas. Comprende: Características, identificadores, Palabras Reservadas, Tipos de Datos, Operadores, Expresiones y Funciones Estándar. Sentencias Básicas: de Asignación, de Entrada y Salida, Sentencias de Bifurcación Condicional, Estructura de Control Repetitivo. Arreglos de Multidimensionales, Funciones y Procedimientos. Archivos de Texto Archivos con Tipo. Diseño de interfase gráfica del usuario: Objetos de diseño. Eventos, cambio de estados de los objetos y principales procedimientos asociados a los eventos Tipo, declaración e Inicialización de Variables: Tipos de Datos, Variables de Tipo. Funciones y procedimientos. Funciones de Entrada y Salida, Funciones matemáticas y funciones para el manejo de caracteres. Sentencias de Control del Programa, Bucle, Arreglos de Cadenas. Creación y mantenimiento de tablas, operaciones de mantenimiento de las tablas a través de la interfase gráfica del usuario y la programación.

III. COMPETENCIAS Y CAPACIDADES

3.1 Competencias

Construye algoritmos que resuelven problemas matemáticos y de actividad humana.

Elabora algoritmos utilizando estructuras lógicas de control secuencial y selectivo.

Aplica estructuras de programación como un lenguaje o medio de comunicación.

Elabora programas , aplicando los conocimientos previos aprendidos, mediante Lenguajes de Programacion.

3.2 Capacidades

Resuelve algoritmos utilizando expresiones con variables y operadores.

Plantea algoritmos utilizando estructuras lógicas de control anidadas y funciones predefinidas.

Construye algoritmos utilizando funciones anidadas y arreglos.

Efectúa programas en el cual aplique todas las estructuras aprendidas.

3.3 Contenidos actitudinales

Comprende contenidos que permitan utilizar los algoritmos asignando variables y constantes.

Utiliza las sentencias de decisión para escoger la opción que corresponde al caso propuesto.

Utiliza las instrucciones de repetición para resolver problemas recursivos.

Expresa los problemas que se presentan en toda actividad a través de programas realizados en Lenguaje de Programación.

I. PROGRAMACIÓN DE CONTENIDOS Y ACTIVIDADES

UNIDAD I : PRINCIPIOS GENERALES

CAPACIDAD: Efectúa técnicas de programación.

SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	TOTAL HORAS
1	Conceptos fundamentales de programación <ul style="list-style-type: none"> • Conceptos generales de computadores y procesamiento de datos. • Los lenguajes de programación • Las herramientas de desarrollo. • Etapas para la elaboración de un programa. 	Resuelve algoritmos utilizando expresiones con variables y operadores.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
2	Tipos de datos, variables y operadores <ul style="list-style-type: none"> • Tipos de Datos • Variables, constantes e identificadores • Expresiones y operadores aritméticos y lógico. 	Resuelve algoritmos utilizando expresiones con variables y operadores.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
3	Estructura lógica de control secuencial <ul style="list-style-type: none"> • Pseudocódigo • Diagrama de flujo • Instrucciones de asignación, entrada y salida 	Construye algoritmos utilizando estructuras lógicas de control secuencial y selectivo.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
4	Estructuras lógicas de control selectivas <ul style="list-style-type: none"> • Estructura selectiva simple y múltiple 	Plantea algoritmos utilizando estructuras lógicas de control secuencial y selectiva.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	5
5	Estructura lógica de control repetitiva para <ul style="list-style-type: none"> • Estructuras Repetitivas para. • Contadores, Acumuladores 	Construye algoritmos utilizando estructuras lógicas de control repetitivas "para" y "mientras".	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 2 hora · Ejercicios en aula - 2 horas 	5

UNIDAD II: SENTENCIAS DE REPETICION

CAPACIDAD: Representa los algoritmos con sentencias de repeticion

SEMAN A	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS
6	Estructura lógica de control repetitiva mientras <ul style="list-style-type: none"> • Estructuras Repetitivas, mientras. 	Construye algoritmos utilizando estructuras lógicas de control repetitivas "para" y "mientras".	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 horas · Ejercicios en aula - 2 horas 	4
7	Estructuras lógicas de control anidadas <ul style="list-style-type: none"> • Estructuras Anidadas • Control de errores 	Construye algoritmos utilizando estructuras lógicas de control anidadas y funciones predefinidas.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 horas · Ejercicios en aula - 2 horas 	4
8	EXAMEN PARCIAL			
9	Funciones predefinidas <ul style="list-style-type: none"> • Conceptos básicos y características • Funciones propias del sistema 	Plantea algoritmos utilizando estructuras lógicas de control anidadas y funciones predefinidas.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 horas · Ejercicios en aula - 2 horas 	4
10	Funciones definidas por el usuario <ul style="list-style-type: none"> • Conceptos básicos y características • Creación de funciones (métodos) de usuario • Argumentos por valor y referencia. 	Construye algoritmos utilizando funciones definidas por el usuario.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 horas · Ejercicios en aula - 2 horas 	4

UNIDAD III: FUNCIONES				
CAPACIDAD : Desarrolla algoritmos con funciones				
SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	TOTAL HORAS
11	Funciones anidadas <ul style="list-style-type: none"> • Utilizando funciones dentro de funciones • Estructuras Anidadas 	Construye algoritmos utilizando funciones anidadas y arreglos.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
12	Introducción a los arreglos <ul style="list-style-type: none"> • Conceptos Básicos • Arreglos Unidimensionales 	Construye algoritmos utilizando funciones anidadas y arreglos.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4

UNIDAD IV APLICACIONES PRACTICAS				
CAPACIDAD: Efectúa programas que manipulan objetos eléctricos.				
SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	TOTAL HORAS

13	Ordenación de arreglos	Construye algoritmos de ordenación y búsqueda.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
14	Ordenación de arreglos <ul style="list-style-type: none"> • Métodos de ordenamiento unidimensionales. 	Construye algoritmos de ordenación y búsqueda.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4
15	Búsqueda en arreglos <ul style="list-style-type: none"> • Búsqueda en arreglos unidimensionales 	Construye algoritmos en el cual aplique todas las estructuras aprendidas.	Lectivas (L): <ul style="list-style-type: none"> · Introducción al tema - 1 hora · Desarrollo del tema – 1 hora · Ejercicios en aula - 2 horas 	4

16	EXAMEN FINAL				
17	EXAMEN SUSTITUTORIO				

II. ESTRATEGIAS DIDÁCTICAS

- Método Expositivo – Interactivo. Disertación docente, participación activa del estudiante.
- Método de Discusión Guiada. Conducción del grupo para abordar situaciones y llegar a conclusiones.
- Método de Demostración – Ejecución. El docente ejecuta para demostrar cómo y con qué se hace y el estudiante ejecuta, para demostrar lo que aprendió.

III. RECURSOS Y MATERIALES

Equipos: proyector de multimedia.
Computadoras.
Software de Programación.

IV. METODOLOGIA

La Facultad de Ingeniería Eléctrica y Electrónica de la UNAC, en cumplimiento con lo dispuesto en la Resolución Viceministerial N°085-2020-MINEDU del 01 de abril de 2020, de manera excepcional y mientras duren las medidas adoptadas por el Gobierno está impartiendo educación remota no presencial haciendo uso de una plataforma virtual educativa utilizando tecnologías de la información y comunicación (TIC). La plataforma virtual de la UNAC es parte del Sistema de Gestión Académico (SGA-UNAC) basado en Moodle, en donde los estudiantes, tendrán a su disposición información detallada del curso: el sílabo, la programación de actividades, material de lectura, instrumentos de evaluación de entregables calificados, y los contenidos de la clase estructurados para cada sesión educativa. La plataforma virtual del SGA será complementada con las diferentes soluciones que brinda Google Suite for Education y otras herramientas tecnológicas multiplataforma como soporte de comunicación tales como Google Meet, Classroom, Google Drive, correo institucional y otros como el ZOOM y MS Team, de ser pertinentes. Las estrategias metodológicas para el desarrollo de las sesiones teóricas y prácticas permiten dos modalidades de aprendizaje en los estudiantes:

MODALIDAD SINCRÓNICA

Forma de aprendizaje basado en el uso de herramientas que permiten la comunicación no presencial y en tiempo real entre el docente y los estudiantes. Dentro de la modalidad sincrónica, se hará uso de: Clases dinámicas e interactivas (virtuales): el docente genera permanentemente expectativa por el tema, a través de actividades que permiten vincular los saberes previos con el nuevo conocimiento, promoviendo la interacción mediante el diálogo y debate sobre los contenidos. Talleres de aplicación (virtuales): el docente genera situaciones de aprendizaje para la transferencia de los aprendizajes a contextos reales o cercanos a los participantes que serán retroalimentados en clase. Tutorías (virtuales): Para facilitar la demostración, presentación y corrección de los avances del informe final de investigación.

MODALIDAD ASINCRÓNICA

Forma de aprendizaje basado en el uso de herramientas que posibilitan el intercambio de mensajes e información entre los estudiantes y el docente, sin interacción instantánea. Dentro de la modalidad sincrónica, se hará uso de: Aprendizaje basado en proyectos (virtual): Permite que el estudiante adquiriera conocimientos y competencias mediante la ejecución de su proyecto de consulta, para dar respuesta a problemas del contexto. Portafolio de evidencias (digital): Permite dar seguimiento a la organización y presentación de evidencias de investigación y recopilación de información para poder observar, contrastar, sugerir, incentivar y preguntar. Foro de investigación (virtual): se realizarán foros de debate, a partir de un reactivo sobre el tema de la sesión de aprendizaje. Trabajos colaborativos (remoto) en plataforma virtual

de aprendizaje. Metodología de búsqueda y administración de información en la web y en ambientes virtuales de aprendizaje.

SOPORTE DE COMUNICACIÓN MULTIPLATAFORMA:

SGA-UNAC, Google Meet, Classroom, ZOOM, Google Drive y correo institucional.

MATERIALES EDUCATIVOS Y OTROS RECURSOS DIDÁCTICOS

Plataforma virtual, usando las herramientas ZOOM y GOOGLE MEET

Equipos multimedia: Laptop, pizarra virtual de las herramientas, etc.

Software de Programación (libre).

V. EVALUACIÓN DEL APRENDIZAJE

De acuerdo con los artículos 82°, 83°, 84° y 85° del Reglamento General de Estudios de la Universidad Nacional del Callao, aprobado con Res. N ° 185-2017-CU, de fecha 27 de junio del 2017, se tendrá a consideración lo siguiente: Participación activa en todas las tareas de aprendizaje. El estudiante aprueba si su Promedio Final es mayor o igual a 10.50 El examen sustitutorio se realizará de acuerdo con la normativa vigente.

CRITERIOS DE EVALUACIÓN: Se evaluará mediante un examen parcial, un examen final más las notas de promedio de prácticas así como (trabajos prácticos dirigidos , examen actitudinal y proyección y responsabilidad social , Adicionalmente se tomará un examen sustitutorio que reemplazará la nota más baja de una de las dos evaluaciones escritas parcial o final.

V. EVALUACIÓN DEL APRENDIZAJE

El promedio final se obtiene del modo siguiente:

EP = Examen Parcial	}	(55%)
EF = Examen Final		
PP = Promedio de Practicas		
PL = Promedio Practicas Dirigidas		(30%)
EA = Examen Actitudinal		(10%)
PRS = Proyección y Responsabilidad Social		(5%)

Nota mínima aprobatoria: 10,5.

VI. FUENTES DE CONSULTA.

5.1 Bibliográficas

Borland C++ manual de referencia	Herbert Schildt Mc Graw - Hill España 1998
C++ Un enfoque orientado a objetos	Luis Joyanes Aguilar Mc Graw – Hill España 1998
Guía de programación en C++	Julio Vásquez Paragulla Universidad San Martín de Porres Perú 1999

5.2 Electrónicas

- ✓ Curso de programación en C++, EUI (UPV), Valencia, 17 al 28 de Julio de 1995 link:
<http://www.uv.es/~sto/cursos/c++/curso95.pdf>
- ✓ Programación en C++ De https://es.wikibooks.org/wiki/Programaci%C3%B3n_en_C%2B%2B
- ✓ Aprende a programar en C++ desde cero De <https://www.programarya.com/Cursos/C++>