

## PROGRAMA DE ASIGNATURA

### ANTECEDENTES GENERALES

Carrera	Ingeniería Civil Eléctrica			
Nombre de la asignatura	Taller de Programacion			
Código de la asignatura	IEIE-25			
Año/Semestre	Primer/Segundo			
Coordinador Académico	Simon Martinez Rozas			
Equipo docente	Simon Martinez Rozas / Mauricio Trigo Gonzales			
Área de formación	General			
Créditos SCT	6			
Horas de dedicación	Actividad presencial	3 horas	Trabajo autónomo	6 horas
Fecha de inicio	Agosto			
Fecha de término	Diciembre			

### DESCRIPCIÓN DE LA ASIGNATURA

Asignatura de naturaleza profesional, obligatoria, teórico-práctica. Domina un cuerpo distintivo y fundamental de conocimientos y técnicas de la ciencia aplicada basado en: **Sistemas de energía eléctrica, Maquinas eléctricas, Electrónica, Teoría de redes, telecomunicaciones y control automático** que le permite resolver problemas asociados al área de la ingeniería eléctrica.

Esta asignatura les permite desarrollar competencia en cálculo, generación de algoritmos de programación y aplicaciones de simulación para resolver problemas y realizar proyectos, durante la carrera, en las diferentes áreas asociadas a eléctricas, comunicaciones y control automáticos.

### RESULTADOS DE APRENDIZAJE

- **Nivel de Desarrollo:**
  - 8.1.1 Identifica los conceptos fundamentales de las ciencias de la ingeniería necesarias para la solución de problemáticas propias del ingeniero de base científica.
- **Resultado de Aprendizaje que tributa:**
  - 8.1.1.9 Reconoce los fundamentos de programación de distintos softwares utilizados en el ámbito de la Ingeniería de base científica en forma lógica.
  - 8.1.1.10 Utiliza distintos softwares para la resolución

### UNIDADES DE APRENDIZAJE

#### UNIDAD I: MatLab

1.1 Que es Programar

1.2 Utilizades de la programación

- 1.3 Programación y la Ingeniería Eléctrica
- 1.4 ¿Por que programar?
- 1.5 Lenguajes de programación
- 1.6 Componentes de Programación
- 1.7 ¿Qué es un Algoritmo?
- 1.8 Introducción a Matlab
- 1.9 Elementos Fundamentales (Matrices)
- 1.10 Aplicaciones
  - 1.11 Entorno de Matlab
  - 1.12 Conceptos de la programación
  - 1.13 Variables en Matlab
  - 1.14 Constantes y funciones incorporadas en Matlab
  - 1.15 Bucles (For, While) en Matlab
  - 1.16 Condicionales en Matlab
  - 1.17 Live Scrip en Matlab
  - 1.18 Regla de Operación en Matlab
  - 1.19 Operación de Matrices y Vectores en Matlab
    - 1.19.1 Arreglos
    - 1.19.2 Matrices
    - 1.19.3 Operaciones
- 1.20 Entradas y salidas de datos
- 1.21 Graficas
  - 1.21.1 2D
  - 1.21.2 3D
- 1.22 Estructura de Selección y de Repetición en Matlab

## **UNIDAD II: Jupiter Notebook**

### 2.1 Introducción a Jupyter Notebook y Fundamentos

- 2.1.1 ¿Qué es Jupyter Notebook?: Usos, ventajas y aplicaciones en ciencia de datos y programación.
- 2.1.2 Instalación y Configuración: Instalación de Jupyter Notebook y presentación de entornos alternativos (como Google Colab).
- 2.1.3 Estructura y Componentes Básicos: Explicación de celdas de código y celdas de texto.
- 2.1.4 Ejecutar Código Básico: Ejecución de celdas, interrupción y reinicio de kernel.
- 2.1.5 Atajos de Teclado y Comandos Básicos: Introducción a atajos esenciales y navegación.

## 2.2 Uso de Markdown y Documentación en Notebooks

2.2.1 Introducción a Markdown: Sintaxis básica de Markdown (títulos, listas, negrita, cursiva).

2.2.2 Documentación de Celdas: Cómo usar celdas de texto para documentar el código y explicar el flujo de trabajo.

2.2.3 Incorporación de Imágenes y Enlaces: Uso de imágenes y enlaces externos para enriquecer los documentos.

2.2.4 Fórmulas Matemáticas con LaTeX: Introducción a la notación matemática básica en Markdown.

## 2.3 Conceptos Básicos de Programación en Python

2.3.1 Variables y Tipos de Datos: Variables, tipos de datos (enteros, flotantes, cadenas) y operadores.

2.3.2 Control de Flujo: Estructuras condicionales (`if`, `else`) y bucles (`for`, `while`).

2.3.3 Funciones Básicas en Python: Definición y uso de funciones (`def`), con ejemplos de funciones sencillas.

2.3.4 Errores Comunes y Depuración Básica: Identificación y solución de errores básicos en Jupyter.

## 2.4 Análisis de Datos con Bibliotecas Básicas

2.4.1 Introducción a Numpy: Uso de Numpy para operaciones básicas y manejo de arrays.

2.4.2 Introducción a Pandas: Carga de datos, creación de DataFrames, selección y filtrado de datos.

2.4.3 Manipulación de Datos: Operaciones de limpieza y manipulación básicas (faltantes, duplicados).

2.4.4 Ejercicio Práctico: Análisis de un dataset simple para aplicar lo aprendido.

## 2.5 Visualización de Datos en Jupyter Notebook

2.5.1 Introducción a Matplotlib: Creación de gráficos básicos (líneas, barras, dispersión).

2.5.2 Uso de Seaborn (Opcional): Introducción a gráficos más avanzados y visualización de relaciones de datos.

2.5.3 Configuración y Personalización de Gráficos: Ajuste de títulos, etiquetas y leyendas.

2.5.4 Ejercicio de Visualización: Crear gráficos de un conjunto de datos y análisis de resultados.

## 2.6 Exportación, Compartición y Buenas Prácticas

2.6.1 Guardar y Exportar Notebooks: Formatos de exportación (HTML, PDF, script Python).

2.6.2 Compartir Notebooks: Opciones de colaboración (GitHub, Google Colab).

2.6.3 Buenas Prácticas en Jupyter Notebook: Organización, comentarios y estructura de los notebooks.

## METODOLOGÍA Y EVALUACIÓN

RESULTADOS DE APRENDIZAJE	*ESTRATEGIA DIDÁCTICA / TÉCNICA DIDÁCTICA	PROCEDIMIENTOS DE EVALUACIÓN
8.1.1.9 Reconoce los fundamentos de programación de distintos softwares utilizados en el ámbito de la Ingeniería de base científica en forma lógica.	<ol style="list-style-type: none"> <li>1. Aprendizaje Basado en Proyectos (ABP) <ul style="list-style-type: none"> <li>• Los estudiantes trabajan en proyectos reales o simulados que requieren la aplicación de conceptos y habilidades de programación.</li> </ul> </li> <li>2. Codificación en Vivo (Live Coding) <ul style="list-style-type: none"> <li>• El profesor programa en tiempo real mientras explica cada paso y toma decisiones de codificación. Los estudiantes observan y hacen preguntas en tiempo real.</li> </ul> </li> <li>3. Retos de Programación (Coding Challenges) <ul style="list-style-type: none"> <li>• Plantea pequeños problemas de programación que los estudiantes deben resolver en un tiempo determinado. Esto fomenta el pensamiento lógico y la aplicación inmediata de lo aprendido.</li> </ul> </li> </ol>	<ul style="list-style-type: none"> <li>• Trabajos individuales</li> <li>• Practicas experimentales</li> <li>• Evaluaciones</li> <li>• Pruebas cortas</li> </ul>
8.1.1.10 Utiliza distintos softwares para la resolución	<ol style="list-style-type: none"> <li>1. Ejemplos Guiados (Guided Examples) <ul style="list-style-type: none"> <li>• El Profesor proporciona ejemplos de código bien comentados que los estudiantes deben analizar y replicar antes de adaptarlos a sus propios proyectos.</li> <li>• Utiliza plataformas en línea que permiten a los estudiantes practicar la codificación en un entorno interactivo y recibir feedback inmediato.</li> </ul> </li> </ol>	<ul style="list-style-type: none"> <li>• Trabajos individuales</li> <li>• Practicas experimentales</li> <li>• Evaluaciones</li> <li>• Pruebas cortas</li> </ul>

\*Los indicadores de logro corresponden al estándar del nivel de aprobación descrito en una matriz de valoración.

1° calificación (N1)		
<b>Tipo de evaluación:</b>	Trabajos en clase (50%)	Prueba presencial (50%)
<b>Contenido:</b>	<ul style="list-style-type: none"> <li>• Pruebas cortas</li> <li>• Obs: promedio de todos los trabajos realizados</li> </ul>	Evaluación: <b>16/ Oct/ 2024 P1</b> <b>18/ Oct/ 2024 P2</b>
<b>Modalidad:</b>	Individual	Individual

2° calificación (N2)		
<b>Tipo de evaluación:</b>	Trabajos en clase (50%)	Prueba presencial (50%)
<b>Contenido:</b>	<ul style="list-style-type: none"> <li>• Pruebas cortas</li> <li>• Obs: promedio de todos los trabajos realizados</li> </ul>	Evaluación: <b>27/Nov/2024 P1</b> <b>29/Nov/2024 P2</b>
<b>Modalidad:</b>	Individual	Individual

**Nota final (NF).**  

$$NF = N1*0.6 + N2*0.4$$

### **NORMATIVA DEL CURSO**

---

- Prohibido el uso distractor de Teléfonos Móviles o dispositivo similar. El estudiante que desee revisar su teléfono lo deberá hacer fuera de la sala.
- Para aprobar la asignatura debe aprobar todos los remediales. La nota de aprobación del remedial es un 4.0, por ende, la nota de aprobación del curso es un 4.0.
- Asistencia mínima obligatoria a la asignatura corresponde al 75%.
- Asistencia mínima obligatoria a actividades practica (AP) es del 100%
- Se permitirá el ingreso a la sala a alumnos que lleguen hasta 15 minutos tarde una vez iniciada la clase.
- Se encuentra estrictamente prohibido copiar en evaluaciones. Esto considera el uso de teléfonos celulares, tablets, computadores, apuntes, cuadernos, libros, etc, que incluya información que el alumno pueda utilizar con fines poco éticos dentro de la evaluación. También se considera como acto indebido de copia, el intercambiar información con compañeros y otras personas durante la evaluación. El alumno que sea sorprendido copiando será evaluado con nota 1.0.

- No se permite el ingreso a evaluaciones (pruebas, talleres, exámenes, remediales, exposiciones, etc) a alumnos que lleguen atrasado, es decir, el ingreso a la sala se realizará hasta la hora que comienza la evaluación.
- Solo pueden rendir los remediales los alumnos que hayan realizado todas las evaluaciones.
- La falta a una evaluación será calificada con nota 1.0\*.
- Solo podrán optar a recuperar la evaluación, los alumnos que justifiquen debidamente su inasistencia.
- La nota máxima que se puede obtener en un remedial es un 4.0

\* **De acuerdo al Reglamento del estudiante de pregrado DE 538-2018 Artículo 29:** La asistencia a trabajos prácticos, laboratorios, prácticas e internados será obligatoria en un 100 % para todos los estudiantes.

Artículo 30: El estudiante que no asista a una evaluación será calificado con la nota mínima (1,0). Sin embargo, podrá ser sometido a otra evaluación especial aquel estudiante que justifique su inasistencia, mediante solicitud presentada para su resolución a la Dirección del Departamento que dicta la asignatura, dentro de los tres días hábiles siguientes al término de la causal que provocó su inasistencia. Los certificados médicos que se presenten deberán ser visados por SEMDA. El Director del Departamento tendrá tres días hábiles para responder la solicitud. En caso de aprobarse la solicitud, la nueva evaluación deberá realizarse antes que finalice el semestre e inicie el período de exámenes. En esta oportunidad los contenidos a evaluar deberán ser los evaluados en la oportunidad en la que el estudiante faltó. Si la inasistencia ha sido al examen en primera oportunidad el estudiante se presentará, automáticamente, al examen en segunda oportunidad. Si el estudiante no asistiera al examen en las dos oportunidades deberá elevar la solicitud fundada y con los respaldos adecuados a la unidad que dicta la asignatura, la que resolverá y estipulará, en su caso, los plazos para regularizar la situación, teniendo derecho a los dos exámenes.

En el Reglamento del Estudiante dice. "Artículo 39: Tratándose de actividades curriculares de planes de estudio por resultados de aprendizaje y demostración de competencias. La Actividad académica (asignatura) se aprobará cuando todos los resultados de aprendizajes sean aprobados por el estudiante, teniendo promedio igual o superior a cuatro (4,0). Los estudiantes de las carreras con planes basados en resultados de aprendizaje y demostración de competencias (rediseñados), que hayan reprobado uno o más de los resultados de aprendizaje de la asignatura tendrán derecho a realizar una actividad de evaluación (examen) en primera y segunda oportunidad. En esta actividad deberá evaluarse el o los resultados de aprendizaje no logrados. Para los estudiantes que deban rendir estas actividades de evaluación (examen), la calificación de la actividad curricular (asignatura) se obtendrá a partir del promedio de las calificaciones obtenidas en cada resultado de aprendizaje. La calificación obtenida en la actividad de evaluación primera o segunda oportunidad reemplazarán a la(s) de los resultados de aprendizaje no aprobados. En el caso en que, una vez realizadas las actividades de evaluación, se repruebe un resultado de aprendizaje, se reprobará la asignatura con calificación obtenida en dicho resultado de aprendizaje".

Se debe justificar la inasistencia a las evaluaciones para tener derecho a rendir los exámenes.

En el Reglamento del Estudiante de Pregrado de la Universidad de Antofagasta, dice "Artículo 39 BIS: Para planes de estudios por objetivos o resultados de aprendizajes, tendrán derecho a rendir la evaluación final (examen...), estudiantes que se hayan presentado a las evaluaciones provistas durante el semestre y cuya nota promedio por cada objetivo de aprendizaje o resultado de aprendizaje sea inferior a 4.0. Esta condición no aplica para estudiantes que se ausentaron a las evaluaciones respectivas. En caso de ausencia a las evaluaciones, y que no aplique al artículo N°30 de este reglamento, se deberá justificar con la unidad académica respectiva, debiendo reprogramarse la evaluación..."

## BIBLIOGRAFÍA.

---

Documentación utilizada para estudio y análisis de curso:

- Apuntes de Clases: Uso de Matlab y Jupyter Notebook
- Web Site Oficial Matlab, sección documentación y ayuda: <https://la.mathworks.com/help/matlab/>
- Web Site Oficial Jupyter Notebook, sección documentación y ayuda: <https://docs.jupyter.org/en/latest/>