**Universidad Americana**

**Ingeniería y Arquitectura**



**Integrantes**

Gabriel Antonio Rojas Uriarte

Valeria Carolina Grijalva Arevalo

Johaneris Sayrin Avalos Fernandez

Manuel Joaquín Chamorro Gomez

**Docente**

**José Andres Munguia Cortez**

**Grupo 4**

**Fecha 07/09/2025**

El proyecto de Álgebra Lineal tiene como objetivo ofrecer una calculadora didáctica en Python para la resolución de sistemas de ecuaciones lineales, tanto de dos como de tres incógnitas, mediante distintos métodos clásicos. Se diseñó con un enfoque pedagógico, utilizando nombres de variables en español y abundantes comentarios para facilitar su comprensión. Además, se emplea la librería fractions.Fraction con el fin de obtener resultados exactos y evitar errores de redondeo, lo que permite mostrar procedimientos claros y confiables.

La estructura del proyecto está organizada en carpetas según los métodos y dimensiones de los sistemas a resolver. En la carpeta sistemas2x2 se encuentra un submódulo robusto que integra varios componentes: parsing.py, encargado de transformar las ecuaciones escritas en texto en estructuras internas válidas, validando fracciones, signos y errores comunes; models.py, que define las clases EcuacionLineal y Sistema2x2, con operaciones exactas y funciones para detectar soluciones únicas, paralelas o infinitas; graph.py, que calcula los puntos de intersección y de corte con los ejes para representar gráficamente las rectas; y main.py, que ofrece una interfaz en consola con dos modos de salida: rápido (que devuelve únicamente la solución y datos básicos para graficar) y paso a paso (que muestra todo el procedimiento algebraico detallado).

Además de este submódulo, existen carpetas dedicadas a los distintos métodos para sistemas de dos ecuaciones. En la carpeta 2x2 se encuentra el archivo igualacion.py, que desarrolla el método de igualación, resolviendo el sistema al despejar las variables y compararlas. La carpeta dos\_por\_dos incluye Funcion\_Sustitucion.py, que implementa el método de sustitución y detalla cada uno de los pasos. En la carpeta Eliminación\_2x2 se localiza Reduccion.py, el cual aplica el método de reducción o eliminación, multiplicando ecuaciones y cancelando variables hasta obtener la solución final.

Para los sistemas de tres ecuaciones, la carpeta 3x3 contiene gauss\_jordan.py, que implementa el método de Gauss–Jordan sobre la matriz aumentada, llevando el sistema a su forma escalonada reducida y determinando si tiene solución única, infinitas soluciones o ninguna. Por su parte, la carpeta tres\_por\_tres ofrece dos alternativas: Funcion\_Eliminacion3x3.py, que aplica eliminación directamente sobre las ecuaciones sin usar matrices explícitas, y Funcion\_Gauss\_jordan.py, que desarrolla también el método de Gauss–Jordan con explicaciones paso a paso. Adicionalmente, la carpeta MetodoGauss incluye GaussBeta.py, una versión experimental o en desarrollo del método de Gauss. Finalmente, la carpeta main contiene mainmenu.py, un menú principal que sirve como punto de entrada para elegir entre los distintos métodos y dimensiones disponibles.

Para ejecutar el proyecto es recomendable crear un entorno virtual de Python, instalar las dependencias listadas en requirements.txt (entre ellas rich y colorama para mejorar la experiencia en consola) y, en caso de que el curso lo requiera, eliminar numpy de la lista, ya que no es esencial. Una vez configurado el entorno, **para iniciar el programa primero debe descargar la carpeta, después debe extraer los archivos del zip, después abre la carpeta en visual studio code, después se va a la carpeta que dice main y ejecuta el código**, para resolver sistemas de dos ecuaciones en modo rápido o paso a paso, o los archivos igualacion.py, Funcion\_Sustitucion.py, Reduccion.py y gauss\_jordan.py para probar cada método por separado.

En cuanto a los formatos de entrada, el programa admite diferentes modalidades según el módulo que se ejecute: algunos solicitan directamente las ecuaciones completas y otros trabajan con los coeficientes y términos independientes. Este enfoque permite adaptarse a distintos estilos de práctica y evita confusiones con las variables. Se recomienda usar fracciones en la forma a/b en lugar de decimales para garantizar exactitud. Además, el programa valida errores como denominadores iguales a cero o entradas inválidas, mostrando mensajes claros en caso de inconsistencias.

En resumen, este proyecto ofrece una solución completa y modular para abordar la enseñanza y práctica de los métodos clásicos de resolución de sistemas de ecuaciones lineales, combinando exactitud matemática, claridad en los pasos y flexibilidad para ejecutar cada método de manera independiente o desde un menú central.