# Documento Explicativo para Sustentación: Módulo de Gramática (GUI)

Este documento explica el funcionamiento del módulo de gramática implementado en Python con interfaz gráfica **Tkinter**, parte del proyecto final de Matemáticas Discretas II.

## 🎯 Objetivo del Módulo

Permitir al usuario:

1. Ingresar una gramática formal: símbolos, producciones.
2. Verificar si una frase pertenece a la gramática.
3. Generar frases válidas derivadas aleatoriamente.
4. Usar una **interfaz gráfica amigable** en lugar de consola.

## 🧱 Estructura del Módulo

### Interfaz gráfica (Tkinter)

* **Entrada de datos:**
  + No terminales (separados por espacio).
  + Terminales (separados por espacio).
  + Símbolo inicial.
  + Producciones (una por línea, formato A -> a B).
* **Acciones disponibles:**
  + Verificar si una frase pertenece.
  + Generar 10 frases aleatorias.
  + Salir y volver al menú principal.
* **Zona de resultados:** Un cuadro ScrolledText muestra los resultados de forma clara.

### Lógica interna

#### construir\_gramatica()

Convierte todos los datos ingresados como texto en una estructura interna de gramática:

{'V': set(), 'T': set(), 'S': '', 'P': dict()}

#### pertenece\_frase(frase\_str, gramatica)

Aplica una búsqueda recursiva (backtracking) para determinar si la frase puede ser derivada desde el símbolo inicial.

* Límite de profundidad para evitar loops.

#### generar\_frases(gramatica, n=10)

Construye frases válidas aplicando producciones aleatorias hasta llegar a una cadena solo de terminales.

## 🧠 Argumentos para la Sustentación

* “Transformé la versión de consola a interfaz gráfica para mejorar la usabilidad.”
* “El sistema recibe las producciones y arma la estructura formal de gramática como diccionario.”
* “Usé backtracking recursivo limitado para verificar pertenencia, como se hace en autómatas.”
* “El botón de generar frases crea hasta 10 derivaciones aleatorias posibles.”
* “Evito loops con control de profundidad y validez final (solo terminales).”
* “Todo resultado es mostrado en una caja desplazable para claridad y seguimiento.”

## 📝 Evaluación según la Rúbrica

| Criterio | Cumplimiento |
| --- | --- |
| Ingreso de gramática | ✔️ Por GUI intuitiva |
| Verificación de frases | ✔️ Con función de backtracking |
| Generación de frases | ✔️ 10 frases aleatorias válidas |
| Uso de interfaz (GUI) | ✔️ Implementado con Tkinter |
| Claridad en sustentación | ✔️ Código modular, fácil de explicar |
| Modularidad / reutilización | ✔️ Funciones puras y separadas |

## ✅ Conclusión

Este módulo es completo, intuitivo, sustentable y cumple con todos los aspectos requeridos por el proyecto. Además, usa principios claros de teoría de lenguajes y está listo para ser defendido con confianza.