



Trabajo Práctico Integrador

Matemática y Programación I



Título del Proyecto

Generador de Tabla de Verdad en Python



Integrante

Sebastián Marcelo Sariella



Descripción del Proyecto

Este proyecto consiste en el desarrollo de un programa en Python que genera tablas de verdad para expresiones booleanas simples. El usuario puede seleccionar entre distintas operaciones lógicas (AND, OR, NOT), y el programa muestra la tabla correspondiente. El sistema permite repetir el proceso tantas veces como se desee, hasta que el usuario decida salir.



Objetivos

- Aplicar conceptos de lógica booleana y programación estructurada.
- Utilizar estructuras secuenciales, condicionales, repetitivas y listas.
- Integrar herramientas de inteligencia artificial en todas las fases del desarrollo.
- Presentar el funcionamiento del programa mediante un video explicativo.



Herramientas de Inteligencia Artificial Utilizadas

Herramienta	Rol en el Proyecto
ChatGPT	Diseño del programa, explicación de conceptos, redacción del documento y guión del video
Copilot	Asistencia en la escritura del código, sugerencias de estructura y corrección de errores
DeepSeek	Revisión del código, mejora de legibilidad y detección de redundancias

Justificación: ChatGPT fue la herramienta más útil para pensar el problema y estructurar el proyecto. Copilot resultó ideal para escribir y completar el código de forma rápida y eficiente. DeepSeek permitió refinar el código y mejorar su calidad.



Código del Programa

```
# -----
# Generador de Tabla de Verdad en Python
# Autor: Sebastián Marcelo Sariella
# Descripción: Este programa permite generar tablas de verdad
# para operaciones lógicas básicas (AND, OR, NOT).
# El usuario puede elegir la operación y repetir el proceso
# hasta que decida salir.
# -----

# Mensaje de bienvenida y opciones disponibles
print("Bienvenido al Generador de Tabla de Verdad")
print("Operaciones disponibles: AND, OR, NOT")
print("Escriba 'SALIR' para terminar el programa.\n")

# Lista con los posibles valores booleanos
valores = [0, 1]

# Bucle principal: se repite hasta que el usuario escriba
# 'SALIR'
while True:
    # Solicita al usuario la operación lógica deseada
    operacion = input("Ingrese la operación lógica que desea
    usar: ").upper()

    # Condición de salida del programa
    if operacion == "SALIR":
        print("Gracias por usar el generador. ¡Hasta la
        próxima!")
        break

    # Generación de la tabla de verdad según la operación
    seleccionada
    print("\nTabla de Verdad:")

    # Operación NOT: solo necesita una variable (A)
    if operacion == "NOT":
        print("A | NOT A")
        for A in valores:
            # NOT invierte el valor: 0 → 1, 1 → 0
            resultado = 1 if A == 0 else 0
            print(f"{A} | {resultado}")
```

```

# Operación AND: requiere dos variables (A y B)
elif operacion == "AND":
    print("A | B | A AND B")
    for A in valores:
        for B in valores:
            # AND solo da 1 si ambos son 1
            resultado = 1 if A == 1 and B == 1 else 0
            print(f"{A} | {B} | {resultado}")

# Operación OR: requiere dos variables (A y B)
elif operacion == "OR":
    print("A | B | A OR B")
    for A in valores:
        for B in valores:
            # OR da 1 si al menos uno es 1
            resultado = 1 if A == 1 or B == 1 else 0
            print(f"{A} | {B} | {resultado}")

# Si la operación ingresada no es válida
else:
    print("Operación no válida. Intente con AND, OR, NOT
o SALIR.")

# Separador visual entre ejecuciones
print("\n-----\n")

```



Conclusión

Este proyecto permitió integrar conocimientos de lógica matemática y programación básica en Python. Además, se logró aplicar herramientas de inteligencia artificial en todas las fases del desarrollo, lo que optimizó el proceso y mejoró la calidad del producto final.