**THERMODYNAMIC CYCLE SIMULATOR**

* Simulates cycles like Rankine,otto,Brayton
* **Inputs:** Compression ratio,temperature,pressure
* **Outputs:** T-S diagram,cycle efficiency,work,output
* **Libraries:** matplotlib,cool prop
* **Extension:** Different working fluids(air,water,refrigerants)

Source Code:

import math

from CoolProp.CoolProp import PropsSI

import matplotlib.pyplot as plt

R\_air = 287.05

gamma\_air = 1.4

def otto\_cycle(r, T1):

    """Simulate ideal Otto cycle given compression ratio and initial temperature (K)."""

    T2 = T1 \* r\*\*(gamma\_air - 1)

    T3 = 2800

    T4 = T3 \* (1 / r)\*\*(gamma\_air - 1)

    efficiency = 1 - 1 / (r\*\*(gamma\_air - 1))

    print("\n--- Otto Cycle ---")

    print(f"Compression ratio: {r}")

    print(f"Efficiency: {efficiency \* 100:.2f}%")

    print(f"T1: {T1:.2f} K, T2: {T2:.2f} K, T3: {T3:.2f} K, T4: {T4:.2f} K")

    s1 = 0

    s2 = s1

    s3 = 1

    s4 = s3

    T\_values = [T1, T2, T3, T4, T1]

    s\_values = [s1, s2, s3, s4, s1]

    def plot\_cycle(T, s, title):

        plt.figure()

        plt.plot(s, T, marker='o')

        plt.xlabel('Entropy (J/kg·K)')

        plt.ylabel('Temperature (K)')

        plt.title(title)

        plt.grid(True)

        plt.show()

    plot\_cycle(T\_values, s\_values, 'Ideal Otto Cycle T-s Diagram')

otto\_cycle(r=8, T1=300)

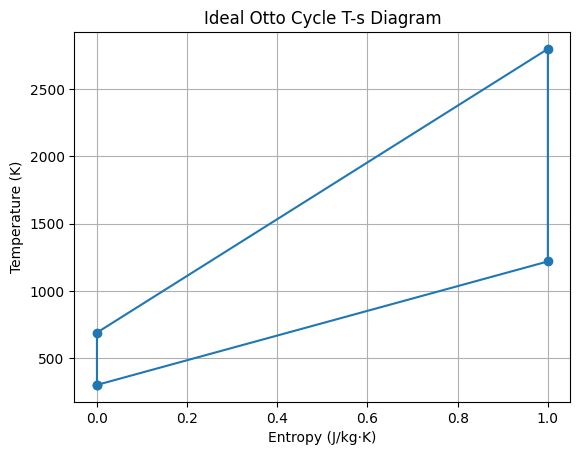
**OUTPUT:**

--- Otto Cycle ---

Compression ratio: 8

Efficiency: 56.47%

T1: 300.00 K, T2: 689.22 K, T3: 2800.00 K, T4: 1218.77 K

****

**CONCLISION:**

**Otto Cycle** (ideal spark-ignition engine) shows distinct isentropic compression and expansion processes with heat addition at constant volume. Its efficiency strongly depends on the compression ratio, improving with higher compression.