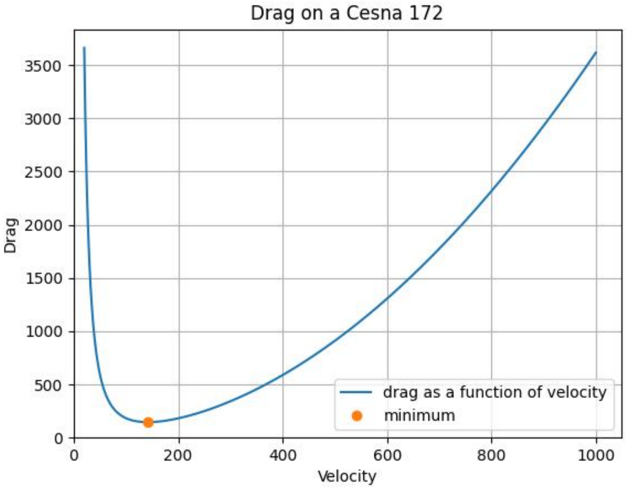
PYTHON CODE

```
from lib.Header import *

"""
Objective: Practice minimization method to find the minimum value of drag force on a Cessna 172.
The airspeed for the minimum drag force is called best range airspeed, Vbr.
Find the best range airspeed, V.
Hint: When an aircraft flies at the best range airspeed, the drag force of the aircraft will be minimal.

Equations for the drag force computations:

Best Range Speed: min(D)
Find Best Range Speed, v_br : min(P/V)=min(DV/V)=min(D)

Input Data (Reference to Cessna 172):
"""
rho = 0.002308  # slug/ft^3, air density at 10,000 ft ALT.
A   = 174       # Wing Wetted Area , ft^2, Cessna 172
W   = 2450      # Gross WT, LB, Cessna 172
AR  = 6.5       # Wing Aspect Ratio
Cdo = 0.018     # Cd_node, Drag Coeff for zero lift
tol = 1e-6      # Tolerance of the minimization

def D(v):
    return (Cd(v)/Cl(v)) * W

def Cd(v):
    return Cdo + ((Cl(v)**2)/(np.pi * AR))

def Cl(v):
    return (W)/(.5 * rho * v**2 * A)


def GoldenSearch(f, a, b, tol):
    R = (-1 + np.sqrt(5)) / 2
    h = b - a
    c = b - h*R
    d = a + h*R
    fc = f(c)
    fd = f(d)

    for i in range(10000):
        if (b - a) < tol:
            break
        if fc > fd:
            a = c
            c = d
            d = a + (b-a)*R
            fc = fd
            fd = f(d)
        else:
            b = d
            d = c
            c = b - (b-a)*R
            fd = fc
            fc = f(c)

    x_min = (a + b)/2
    f_min = f(x_min)

    return [x_min, f_min]


velocity = np.linspace(20, 1000, 10000)
[vmin, fmin] = GoldenSearch(D, 20, 1000, tol)
print(vmin, fmin)

plt.figure()
plt.plot(velocity, D(velocity), label='drag as a function of velocity')
plt.plot(vmin, fmin, 'o', label='minimum')
plt.legend()
plt.xlim(0)
plt.ylim(0)
plt.title("Drag on a Cesna 172")
plt.xlabel("Velocity (ft/s)")
plt.ylabel("Drag")
plt.grid()
SAVE(1)
plt.show()

PDF("Assignment3.py", "Assignment3.pdf")
```

OUTPUT

Plots



Prints