```python

# coding: utf-8

__author__ = 'Geoffrey Nyaga'

import sys
sys.path.append('../')
from API.db_API import write_to_db,read_from_db

from math import sqrt
import numpy as np
import matplotlib.pyplot as plt

import API.perfIntroAPI as pfintro

altitude = 8500 #ft
rhoSL  = read_from_db('rhoSL')
gamma = 1.4   #do sth
gas_constant = 1716
cruiseSpeed = read_from_db('cruiseSpeed')
finalMTOW = read_from_db('finalMTOW')
maxSpeed = read_from_db('maxSpeed')
S = read_from_db('S') * 10.764
stallSpeed = read_from_db('stallSpeed')

atmosphere = pfintro.atmosphere(altitude,rhoSL)
# print(atmosphere.pressure(),"psf")
# print(atmosphere.pressureRatio(),"pressure ratio")
# print(atmosphere.density(),"slugs/ft^3")
# print(atmosphere.densityRatio(),"density ratio")
# print(atmosphere.temperature(),"degree R")
# print(atmosphere.temperatureRatio(),"temperature Ratio")

altitude_pressure = atmosphere.pressure()
altitude_density = atmosphere.density()
altitude_temperature = atmosphere.temperature()
altitude_pressureRatio = atmosphere.pressureRatio()

speeds = pfintro.speeds(altitude,altitude_temperature,altitude_pressure,cruiseSpeed,gas_constant,gamma,altitude_density,
    altitude_pressureRatio,rhoSL)
# print(speeds.soundSpeed(),"ft/s Speed of Sound")
# print(speeds.dynamicPressure(),"psf dynamic pressure")
# print(speeds.EAS(),"ft/s equivalent airspeed")
# print(speeds.CAS(),"ft/s calibrated airspeed")

# negloadFactor = -1.2
negCLmin = -1

flightEnvelope = pfintro.flightEnvelope(finalMTOW,S,maxSpeed,stallSpeed,negCLmin,rhoSL)

# print(flightEnvelope.loadFactor(),"load factor")
# print(flightEnvelope.negloadFactor(),"neg load factor")
# print(flightEnvelope.minCruiseSpeed(),"min Cruise Speed")
# print(flightEnvelope.maxCruiseSpeed(),"max Cruise Speed")
# print(flightEnvelope.diveSpeed(),"dive Speed")
# print(flightEnvelope.maneuveringSpeed(),"maneuvering Speed")
print(flightEnvelope.negmaneuveringSpeed(),"neg maneuvering Speed") # AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

maneuveringSpeed = flightEnvelope.maneuveringSpeed()
# CLmax = 1.6 ####flaps down
CLmax = (2*finalMTOW)/(rhoSL*(stallSpeed*1.688)**2*S)
negstallSpeed = sqrt((2*finalMTOW)/(abs(negCLmin)*rhoSL*S))/1.688
print(negstallSpeed,"negstallSpeed") #AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA In the future use -Clmax for neg stall speed

vtop = np.arange(stallSpeed,flightEnvelope.maneuveringSpeed())
# vtop = np.arange(1,150)

vbottom = np.arange(negstallSpeed,flightEnvelope.negmaneuveringSpeed())

xVA = flightEnvelope.maneuveringSpeed()
yVA = 0.003388*xVA**2*S*CLmax / finalMTOW

xVS = stallSpeed
yVS = 0.003388*xVS**2*S*CLmax / finalMTOW

```

```
xnegVS = negstallSpeed
ynegVS = 0.003388*xnegVS**2*S*negCLmin / finalMTOW

linev = 0.003388*vtop**2*S*CLmax / finalMTOW
linevneg = 0.003388*vbottom**2*S*negCLmin / finalMTOW
# plt.scatter(np.linspace(30, 200, 3), np.random.randn(3))
plt.scatter(np.linspace(stallSpeed, flightEnvelope.maneuveringSpeed(), 2), (yVS,yVA))

plt.plot (vtop,linev)
plt.plot (vbottom,linevneg)
plt.plot([flightEnvelope.maneuveringSpeed(),flightEnvelope.diveSpeed()],[flightEnvelope.loadFactor(),flightEnvelope.loadFactor()],color='k',marker='o')
plt.plot([flightEnvelope.diveSpeed(),flightEnvelope.diveSpeed()],[flightEnvelope.negloadFactor(),flightEnvelope.loadFactor()],color='k',marker='o')
plt.plot([flightEnvelope.negmaneuveringSpeed(),flightEnvelope.diveSpeed()],[flightEnvelope.negloadFactor(),flightEnvelope.negloadFactor()],color='k',marker='o')
plt.plot([negstallSpeed,negstallSpeed],[0,ynegVS],color='k',marker='o')

plt.plot([stallSpeed,stallSpeed],[0,yVS],color='k',marker='o')

plt.axhline(y=0,color='k',linewidth=1.0)
plt.axhline(y=flightEnvelope.loadFactor(),linestyle='dashed')
plt.axhline(y=flightEnvelope.negloadFactor(),linestyle='dashed')


plt.title('V-n Diagram')
plt.xlabel('Speed EAS')
plt.ylabel('Load Factor, n')
if __name__ == '__main__':
    plt.show()
```