```
clear;
clc;
syms x

%% Design Parameters
Pas=110; %Number of Passengers
W_C=2000; %Weight of Cargo in pounds
R_SA=2500; %Range in still air in Nautical Miles
TOFL=6000; %TOFL in feet, Sea Level Hot Day 84F
V_App=130; %Knots, Design Payload plus 100% Max Fuel
M_Cr=0.8; %Cruise Mach Number
H_0Cr=35000; %Initial Cruise Altitude in Feet
T_SL=543.67; %Rankine, Sea Level Temperature on a Hot Day

%% Changing Values
AR=8;
AirfoilType=2; % 1 for Conventional 2 for Supercritical
num_eng=2; % Number of Engines

%% Conditions at Cruise Altitude
    if (82300 > H_0Cr) && (H_0Cr > 36150) %Isothermal Region
    Temperature= 389.99;
    PressureRatio=exp(-(32.17/(1718*389.99))*(Altitude));
    DensityRatio=exp(-(32.17/(1718*389.99))*(Altitude));
    Pressure=PressureRatio*472.787;
    Density=DensityRatio*.000705438;
    Viscos=(.317*(Temperature^1.5)*(734.7/(Temperature+216)))/(10^10);
    else
    Temperature= 518.69 - (H_0Cr*.00356); %Gradient Region
    PressureRatio=(Temperature/518.69)^(-32.17/(-.00356*1718));
    DensityRatio=(Temperature/518.69)^(-(32.17/(-.00356*1718)+1));
    Pressure=PressureRatio*2116.2;
    Density=DensityRatio*.0023769;
    Viscos=(.317*(Temperature^1.5)*(734.7/(Temperature+216)))/(10^10);
    end

%% Airplane Configurations

%AirfoilType=1; % 1 for Conventional 2 for Supercritical
%num_eng=3; % Number of Engines
SFC_JT8D=0.78;
SFC_JT9D=0.61;
TR=0.35; % Taper Ratio
%AR=9;
Sweep=35;   %
C_L=0.47; % First CL assumption
Abreast=5; % Seats abreast
ClassInt=1; % 1 for Small 2 for 3 class international
NAisle=1; % Number of Aisles
NFCrew=2; % Number of Flight Crew
NStew=3; % Number of Stewardess

%% Density Ratio at hot day condtion
P_SL=2116.2;
D_SLHD=P_SL/(1718*T_SL);
DensityRatioHD=D_SLHD/.0023769;
```

```matlab
%% Density & Pressure Ratio for Climb
H_Climb=20000;
TemperatureCL= 518.69 - (H_Climb*.00356);
PressureRatioCL=(TemperatureCL/518.69)^(-32.17/(-.00356*1718));
DensityRatioCL=(TemperatureCL/518.69)^(-(32.17/(-.00356*1718)+1));

%% Speed of Sound
a = sqrt(1.4*1718*Temperature);
a_HD = sqrt(1.4*1718*T_SL); % Hot Day Condition

%% Modern vs 1970 Airplane Technology Differences

AdvTech = 1; % 1 for Modern 2 for 1970s

if AdvTech == 1
    ADV_SFC = .9;
    ADV_Eng = 1.1;
    ADV_Wei = 1.1;
else
    ADV_SFC = 1;
    ADV_Eng = 1;
    ADV_Wei = 1;
end




%% Wing Loading
if AirfoilType == 1
    del_MDiv = solve(C_L == -1286.6*x^4 - 54.864*x^3 - 4.7454*x^2 - 2.9731*x
+ 0.549);
    del_MDiv=double(del_MDiv(2));
    MDiv=(M_Cr+.004)-del_MDiv; % Divergence Mach Number

if Sweep == 0
    t_c = -0.7341*MDiv^4 + 2.1504*MDiv^3 - 2.3268*MDiv^2 + 0.4702*MDiv +
0.3783;
elseif Sweep == 10
    t_c = -13.171*MDiv^4 + 37.216*MDiv^3 - 39.294*MDiv^2 + 17.758*MDiv -
2.6468;
elseif Sweep == 15
    t_c = 0.4634*MDiv^4 - 0.6265*MDiv^3 - 0.0328*MDiv^2 - 0.2659*MDiv +
0.4421;
elseif Sweep == 20
    t_c = -3.3562*MDiv^4 + 9.3785*MDiv^3 - 9.7938*MDiv^2 + 3.9651*MDiv -
0.2466;
elseif Sweep == 25
    t_c = -5.9435*MDiv^4 + 17.37*MDiv^3 - 18.977*MDiv^2 + 8.6544*MDiv -
1.1434;
elseif Sweep == 30
    t_c = -1.4073*MDiv^4 + 4.1138*MDiv^3 - 4.484*MDiv^2 + 1.6564*MDiv +
0.1169;
elseif Sweep == 35
    t_c = 3.4499*MDiv^4 - 9.8401*MDiv^3 + 10.487*MDiv^2 - 5.417*MDiv +
1.3583;
```

```
elseif Sweep == 40
    t_c = -2.7698*MDiv^4 + 8.4378*MDiv^3 - 9.5934*MDiv^2 + 4.3958*MDiv -
0.4409;
end

elseif AirfoilType == 2
    del_MDiv = solve(C_L == -2E09*x^6 - 1E08*x^5 - 2E06*x^4 - 2169.3*x^3 +
128.09*x^2 - 5.9018*x + 0.5458);
    del_MDiv=double(del_MDiv(2));
    MDiv=(M_Cr+.004)-del_MDiv; % Divergence Mach Number

if Sweep == 0
    t_c = 221.36*MDiv^4 - 687.39*MDiv^3 + 802.15*MDiv^2 - 417.56*MDiv +
82.018;
elseif Sweep == 5
    t_c = 289.94*MDiv^4 - 882.55*MDiv^3 + 1010*MDiv^2 - 515.79*MDiv + 99.4;
elseif Sweep ==10
    t_c = 50.248*MDiv^4 - 169.18*MDiv^3 + 214.72*MDiv^2 - 122.19*MDiv +
26.446;
elseif Sweep == 15
    %t_c = 146.48*MDiv^4 - 464.65*MDiv^3 + 554.96*MDiv^2 - 296.35*MDiv +
59.903;
    t_c= -22.34*MDiv^3 + 54.632*MDiv^2 - 45.089*MDiv + 12.635;
elseif Sweep == 20
    t_c = 145.25*MDiv^4 - 464.22*MDiv^3 + 558.61*MDiv^2 - 300.57*MDiv +
61.232;
elseif Sweep == 25
    t_c = 97.391*MDiv^4 - 327.95*MDiv^3 + 415.93*MDiv^2 - 236*MDiv + 50.733;
elseif Sweep == 30
    t_c = 2.8761*MDiv^4 - 26.508*MDiv^3 + 57.418*MDiv^2 - 47.825*MDiv +
14.042;
elseif Sweep == 35
    t_c = 1016.4*MDiv^4 - 3450.8*MDiv^3 + 4397.3*MDiv^2 - 2493.5*MDiv +
531.18;
elseif Sweep == 40
    t_c = 4195.8*MDiv^4 - 14912*MDiv^3 + 19874*MDiv^2 - 11772*MDiv + 2615.7;
end
end

A=(cosd(Sweep))^2*(t_c)^2*AR; % Design Parameter given with different Sweep
and AR

C_LMaxTO = 836.2*A^4 - 370.27*A^3 + 20.78*A^2 + 10.596*A + 1.1993;     % Both
C_L values from Figure 3
C_LMaxLDG = -324.81*A^4 + 279.63*A^3 - 99.021*A^2 + 18.894*A + 1.9413;

% Wing Loading at Landing
W_SLDG=(V_App/1.3)^2*(DensityRatioHD*C_LMaxLDG)/296;

V_Cr=M_Cr*a/1.69; % Cruise Velocity in Knots
R_AO=R_SA+200+.75*V_Cr; % All out Range

%Ratio of Weight for JT8D
R_JT8D=solve(R_AO == 102288*x^4 - 75944*x^3 + 34522*x^2 + 4317.4*x + 38.787);
R_JT8D=double(R_JT8D(2));
```

```
%Ratio of Weight for JT9D
R_JT9D=R_JT8D*(SFC_JT9D/SFC_JT8D);
%R_JT9D=.390;

%Wing Loading at Takeoff
%W_STO=W_SLDG/(1-.75*R_JT9D);
W_STO=W_SLDG;

%Initial Cruise Wing Loading
W_SIC=.965*W_STO;

C_LIC=W_SIC/(1481*PressureRatio*M_Cr^2);

% Lift Coefficient Iteration
% if C_L - C_LIC < 0
%     if abs(C_L - C_LIC) > .01
%         C_L = C_L +.01;
%     elseif abs(C_L - C_LIC) <= .01
%    C_L = C_L+.001;
%    end
 %
 %else
 %     if abs(C_L - C_LIC) > .01
 %        C_L = C_L -.1;
 %      elseif abs(C_L - C_LIC) <= .01
 %   C_L = C_L-.001;
 %      end
 %end
%% TOFL
if num_eng == 2
    %B = solve(TOFL*10^-3 == 2E-10*x^4 - 2E-07*x^3 + 7E-05*x^2 + 0.0231*x +
1.0528);
    %B = solve(TOFL*10^-3 == -4E-08*x^3 + 4E-05*x^2 + 0.0259*x + 1.0032);
    B = solve(TOFL*10^-3 == 1E-05*x^2 + 0.03*x + 0.7872);
    B = double(B(2));
elseif num_eng == 3
    B = solve(TOFL*10^-3 == 1E-05*x^2 + 0.0275*x + 0.6443);
    B = double(B(1));
elseif num_eng == 4
    B = solve(TOFL*10^-3 ==1E-05*x^2 + 0.026*x + 0.4268);
    B = double(B(2));
end

WT_7VLO = (B/W_STO)*C_LMaxTO*DensityRatioHD;

V_LO = 1.2*sqrt(296*W_STO/(DensityRatioHD*C_LMaxTO));

%M_LO = V_LO/(a_HD/1.69)/sqrt(DensityRatioHD);
M_LO= (V_LO*sqrt(DensityRatioHD))/(661);

M_7VLO = .7*M_LO;

%JT9D
T_SLST=45500;
```

```
% From JT9D Graph
T_M= -82115*(M_7VLO)^4 + 59015*(M_7VLO)^3 + 28712*(M_7VLO)^2 - 48441*M_7VLO +
45621;
%T_M = 39353*M_7VLO^2 - 48341*M_7VLO + 45573;

W_T = WT_7VLO*T_M/T_SLST;

%% Weight

n=1.5*2.5; %Ultimate Load Factor

Material = 1; % 1 for Composite 2 for Aluminum

if Material == 1

%Wing Weight
if num_eng == 2
    k_w= 1; % Both on Wing
elseif num_eng == 3
    k_w = 1.01; % 2 on Wing 1 on Fuselage
elseif num_eng == 4
    k_w = 1; % All 4 on Wing
end


W_W=(0.00945*(AR^.8)*(1+TR)^.25*k_w*3.75^.5)/((t_c+.03)^.4*cosd(Sweep)*W_STO^
.695);

%Fuselage Weight

if ClassInt == 1
    L=3.76*(Pas/Abreast)+33.2;
    D=1.75*Abreast+1.58*NAisle+1;
elseif ClassInt ==2
    L=(3.76*(Pas/Abreast)+33.2)*1.1;
    D=(1.75*Abreast+1.58*NAisle+1)*1.1;
end

W_F=.6727*11.5*L^.6*D^.72*n^.3;

%Landing Gear
W_LG= 0.04;

%Nacelles and Pylon(s)
W_NP=0.0555/W_T;

%Tail Surface

k_TS=(.08/num_eng)+.17;
W_TS=k_TS*W_W;
%W_TSW=(1+k_TS)*W_W;

%Power Plant
W_PP=1/(3.58*W_T)*ADV_Wei;
```

```
%Fuel
W_Fuel=1.0275*R_JT9D;

%Payload
W_PL=215*Pas+W_C;

%Fixed Equipment
W_FE=132*Pas+300*num_eng+260*NFCrew+170*NStew;

C_A=(W_TS+W_W)*.7;
C_B=W_F*.85;
C_C=W_LG+W_PP+W_Fuel+.035-1+(W_NP*.8);
C_D=W_PL+(W_FE)*.9;

%Takeoff Weight Calculation
 W_TO=107200; % W_TO Assumption
 W_TOC = (C_A)*(W_TO^1.195)+C_B*(W_TO^.235)+(C_C)*W_TO+(C_D)



elseif Material == 2


%Wing Weight
if num_eng == 2
    k_w= 1; % Both on Wing
elseif num_eng == 3
    k_w = 1.01; % 2 on Wing 1 on Fuselage
elseif num_eng == 4
    k_w = 1; % All 4 on Wing
end

W_W=(0.00945*(AR^.8)*(1+TR)^.25*k_w*3.75^.5)/((t_c+.03)^.4*cosd(Sweep)*W_STO^
.695);

%Fuselage Weight

if ClassInt == 1
    L=3.76*(Pas/Abreast)+33.2;
    D=1.75*Abreast+1.58*NAisle+1;
elseif ClassInt ==2
    L=(3.76*(Pas/Abreast)+33.2)*1.1;
    D=(1.75*Abreast+1.58*NAisle+1)*1.1;
end

W_F=.6727*11.5*L^.6*D^.72*n^.3;

%Landing Gear
W_LG= 0.04;

%Nacelles and Pylon(s)
W_NP=0.0555/W_T;
```

```
%Tail Surface
%if num_eng == 2
%    k_TS= .17;
%elseif num_eng == 3
%    k_TS = (.17+.17+.25)/3;
%elseif num_eng == 4
%    k_TS = .17;
%end

k_TS=(.08/num_eng)+.17;
W_TS=k_TS*W_W;
%W_TSW=(1+k_TS)*W_W;

%Power Plant
W_PP=1/(3.58*W_T)*ADV_Wei;

%Fuel
W_Fuel=1.0275*R_JT9D;

%Payload
W_PL=215*Pas+W_C;

%Fixed Equipment
W_FE=132*Pas+300*num_eng+260*NFCrew+170*NStew;

C_A=(W_TS+W_W)*.94;
C_B=W_F*.94;
C_C=W_LG+(W_NP)+W_PP+W_Fuel+.035-1;
C_D=W_PL+(W_FE);

%Takeoff Weight Calculation
 W_TO=137500; % W_TO Assumption
 W_TOC = (C_A)*(W_TO^1.195)+C_B*(W_TO^.235)+(C_C)*W_TO+(C_D)

end

S=W_TO/W_STO;
b=sqrt(AR*S);
MAC=S/b;
T=W_TO/W_T;
TC=T/num_eng; % Thrust per Engine

%% Drag

RN_k=Density*V_Cr/Viscos; % At cruise condition

%Wing
RN_W=RN_k*MAC;
cf_W=0.0449*(RN_W)^-0.159;
c_r=1.5*MAC*(1/(1+TR-(TR/(1+TR))));
c_y=c_r+c_r*(TR-1)*(D/b); %Spanwise distribution of chord
S_WETW=2*(S-D*c_y)*1.02;
```

```
if Sweep == 0 || Sweep <=10
    K_WW = 6.0425*(t_c)^2 + 1.5004*t_c + 1.0045;
elseif Sweep == 15
    K_WW = 6.068*(t_c)^2 + 1.4011*t_c + 1.0043;
elseif Sweep == 20
    K_WW = 6.0023*t_c^2 + 1.2611*t_c + 1.0087;
elseif Sweep == 25
    K_WW =  6.0844*t_c^2 + 1.1591*t_c + 1.0092;
elseif Sweep == 30
    K_WW = 6.0873*t_c^2 + 1.035*t_c + 1.013;
elseif Sweep == 35
    K_WW = 5.973*t_c^2 + 0.9425*t_c + 1.0104;
elseif Sweep == 40
    K_WW = 5.7801*t_c^2 + 0.8708*t_c + 1.0103;
end

f_Wing=K_WW*cf_W*S_WETW;

%Fuselage
RN_F=RN_k*L;
cf_F=0.0449*(RN_F)^-0.159;
S_WETF=.9*pi*D*L;
L_DF=(L/D);
K_WF= 0.0003*(L_DF)^4 - 0.0107*(L_DF)^3 + 0.1314*(L_DF)^2 - 0.7552*(L_DF) +
2.9133;
f_Fuse=K_WF*cf_F*S_WETF;

%Tail
if num_eng == 2
    E_W=2;
    E_F=0;
elseif num_eng == 3
    E_W=2;
    E_F=1;
elseif num_eng == 4
    E_W=4;
    E_F=0;
end
f_Tail=(.35*E_W+.45*E_F)/num_eng*f_Wing;

%Nacelles
S_WETN=2.1*(TC^.5)*num_eng;
f_Nac=1.25*cf_W*S_WETN;

%Pylons
f_Pylon=.2*f_Nac;

%Total
f_Total=(f_Wing+f_Tail+f_Fuse+f_Nac+f_Pylon)*1.06;

%Drag
C_DO=f_Total/S;

e=1/(1.035+(.38*C_DO*pi*AR));
```

```
%% Climb

W_AvgClimb= (1+.965)/2*W_TO;

V_CL=1.3*(12.9/((f_Total*e)^.25))*(W_TO/(DensityRatioCL*b))^.5;

M_Climb= V_CL*1.69/a;

%Thrust Required for Level Flight, CLIMB CONDITION
T_RCL=((DensityRatioCL*f_Total*V_CL^2)/296)+(94.1/(DensityRatioCL*e))*(W_AvgC
limb/b)^2*(1/V_CL^2);

%Thrust Available at climb condition
T_AJT9D=15400;
c_AJT9D=.65;

T_A=(TC/T_SLST)*T_AJT9D;

%Rate of Climb
R_Climb=101*(num_eng*T_A-T_RCL)*V_CL/W_AvgClimb;

%Time to Climb
Time_CL=H_0Cr/R_Climb;

%Range to Climb
Range_CL=V_CL*Time_CL/60;

%Weight of Fuel to Climb
W_FuelCL=c_AJT9D*num_eng*T_A*Time_CL/60*ADV_SFC;

%% Range

W_O=W_TO-W_FuelCL;
W_L=(1-R_JT9D)*W_TO;

C_LAvgCr=((W_O+W_L)/(2*S))/(1481*PressureRatio*M_Cr^2);

C_Di=C_LAvgCr^2/(pi*AR*e);

%Total Drag
% Here we assumed Compressibility Drag = .001
C_DTotal=C_DO+C_Di+.001;

Lift_Drag= C_LAvgCr/C_DTotal;

%Thrust Required for Range
T_RR=((W_O+W_L)/2)/(Lift_Drag);

%Thrust required
T_RRJT9D=T_RR*T_SLST/TC;
T_RRJT9DE= T_RRJT9D/num_eng; % Thrust per Engine

% From JT9D @ 35k Altitude
```

```
% This value will always be around 0.61 - 0.64 since Cruise Mach is given
% as one of the design parameters
if T_RRJT9DE > 6000 && T_RRJT9DE < 6500
    c=0.605;
elseif T_RRJT9DE <= 6000 && T_RRJT9DE > 5500
    c=0.62;
elseif T_RRJT9DE > 6500 && T_RRJT9DE < 9000
    c=0.61;
elseif T_RRJT9DE <5500 && T_RRJT9DE > 5000
    c=0.63;
elseif T_RRJT9DE > 9000 && T_RRJT9DE < 10000
    c=0.63;
end
c_Cruise=.62;
%Cruise Range
R_Cruise=(V_Cr/c_Cruise)*(Lift_Drag)*log(W_O/W_L);

R_Sum=Range_CL+R_Cruise;

if R_Sum > R_AO
    disp('             R_Sum > R_AO               ERROR')
else
    disp('             R_Sum < R_AO             No Error')
end


%% Check for Thrust Required at top of Climb
% Thrust Available Max Cruise = 10000
C_LICC=(W_O/S)/(1481*PressureRatio*M_Cr^2);

C_DiC=C_LICC^2/(pi*AR*e);

C_DC=C_DO+C_DiC+.001; % .001 being the compressibility drag

Lift_DragC=C_LICC/C_DC;

T_ReqC=W_O/Lift_DragC;

T_ReqCE=T_ReqC/num_eng; % Thrust Required Per Engine

T_RJT9DC=T_ReqCE*T_SLST/TC;




%% Climb Gradient

% 1st Segment
C_LTO=C_LMaxTO/(1.2^2);
R_TOTOM=C_LTO/C_LMaxTO;
Del_FCDo= -0.4417*(R_TOTOM)^5 + 2.4979*(R_TOTOM)^4 - 2.1189*(R_TOTOM)^3 +
1.086*(R_TOTOM)^2 - 0.7384*(R_TOTOM) + 0.331;

% Assume Landing gear drag = C_DO
```

```
C_DFCl=C_DO+(.1*Del_FCDo)+C_DO+(C_LTO)^2/(pi*AR*e);

L_DFCl=C_LTO/C_DFCl;

T_ReqFCl=W_TO/L_DFCl;

T_TOFCL = -82115*(M_LO)^4 + 59015*(M_LO)^3 + 28712*(M_LO)^2 - 48441*M_LO +
45621;

T_AEngFCl=TC/T_SLST*T_TOFCL;

GradFCl= (2*T_AEngFCl-T_ReqFCl)/W_TO*100 % 0 Required
if num_eng == 2
    if GradFCl > 0
        disp('                No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
elseif num_eng == 3
    if GradFCl > .3
        disp('                No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
elseif num_eng == 4
    if GradFCl > .5
        disp('                No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
end

% 2nd Segment

C_DSCl=C_DO+(.1*Del_FCDo)+(C_LTO)^2/(pi*AR*e);

L_DSCl=C_LTO/C_DSCl;

T_ReqSCl=W_TO/L_DSCl;

T_AEngSCl = T_AEngFCl;

GradSCl=(2*T_AEngSCl - T_ReqSCl)/W_TO*100 % 2.4% Required
if num_eng == 2
    if GradSCl > 2.4
        disp('                No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
elseif num_eng == 3
    if GradSCl > 2.7
        disp('                No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
```

```
elseif num_eng == 4
    if GradSCl > 3
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
end
% 3rd Segment

% For 0 Sweep
C_LCleanO= -311.11*(t_c)^3 + 79.844*(t_c)^2 - 2.1653*(t_c) + 0.9374;
cor = 2E-05*Sweep^2 - 0.0037*Sweep + 1.1; %corrects cl clean graph for sweep
C_LClean = C_LCleanO-(1.1-cor);

D_HDCl=0.925; % Density Ratio for hot day at 1000 ft
V_TCl=1.2*(296*W_STO/(D_HDCl* C_LMaxTO))^.5;

a_HDCl=659; % Speed of Sound for hot day at 1000 ft
M_TCl=V_TCl/a_HDCl;
C_LTCl=C_LClean/(1.2^2);
C_DTCl=C_DO+(C_LTCl)^2/(pi*AR*e);
L_DTCl=C_LTCl/C_DTCl;
T_ReqTCl=W_TO/L_DTCl;

T_TOTCl= -21428*(M_TCl)^3 + 43382*(M_TCl)^2 - 43523*(M_TCl) + 37935;


T_AEngTCl=TC/T_SLST*T_TOTCl;

GradTCl= ((2*T_AEngTCl - T_ReqTCl)/W_TO)*100 % 1.5% Required
if num_eng == 2
    if GradTCl > 1.2
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
elseif num_eng == 3
    if GradTCl > 1.5
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
elseif num_eng == 4
    if GradTCl > 1.7
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
end

%Approach

C_LApp=C_LMaxTO/(1.3^2);
R_TOTOMApp=1/(1.3^2);
```

```
Del_CDoApp= -0.4417*(R_TOTOMApp)^5 + 2.4979*(R_TOTOMApp)^4 -
2.1189*(R_TOTOMApp)^3 + 1.086*(R_TOTOMApp)^2 - 0.7384*(R_TOTOMApp) + 0.331;

C_DClApp=C_DO+(.1*Del_CDoApp)+(C_LApp^2)/(pi*AR*e);

L_DClApp=C_LApp/C_DClApp;

W_LDGApp=W_SLDG*S;

T_ReqApp=W_LDGApp/L_DClApp;

V_ClApp=(296*W_SLDG/(DensityRatioHD* C_LApp))^.5;

M_ClApp=(V_ClApp*sqrt(DensityRatioHD))/(661);

T_TOClApp = -21428*(M_ClApp)^3 + 43382*(M_ClApp)^2 - 43523*(M_ClApp) + 37935;

T_AReqApp=TC/T_SLST*T_TOClApp;

GradClApp=((2*T_AReqApp - T_ReqApp)/W_LDGApp)*100
if num_eng == 2
    if GradClApp > 2.1
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
elseif num_eng == 3
    if GradClApp > 2.4
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
elseif num_eng == 4
    if GradClApp > 2.7
        disp('               No Grad Error      ')
    else
        disp(            ' GRAD ERROR                ')
    end
end

%Landing

C_LLDG=C_LMaxLDG/(1.3^2);
R_CLLDG=(1/1.3^2);
Del_CDoLDG = 1.8759*(R_CLLDG)^5 - 3.3868*(R_CLLDG)^4 + 2.8764*(R_CLLDG)^3 -
0.4174*(R_CLLDG)^2 - 0.6509*(R_CLLDG) + 0.4135;

C_DClLDG=C_DO+(.1*Del_CDoLDG)+C_DO+(C_LLDG^2)/(pi*AR*e);

L_DClLDG=C_LLDG/C_DClLDG;

T_ReqClLDG=W_LDGApp/(L_DClLDG);
M_LDG=(V_App*sqrt(DensityRatioHD))/(661);
```

```
T_ClLDG = -82115*(M_LDG)^4 + 59015*(M_LDG)^3 + 28712*(M_LDG)^2 -
48441*(M_LDG) + 45621;
T_AReqLDG=TC/T_SLST*T_ClLDG;

GradLDG=(3*T_AReqLDG - T_ReqClLDG)/W_LDGApp*100
if num_eng == 2
    if GradLDG > 3.2
        disp('               No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
elseif num_eng == 3
    if GradLDG > 3.2
        disp('               No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
elseif num_eng == 4
    if GradLDG > 3.2
        disp('               No Grad Error      ')
    else
        disp(             ' GRAD ERROR                ')
    end
end

%% Direct Operating Cost

% Block Speed
% All Times are given in hours & All distance in statute miles
D_Block=R_SA*1.15;
T_GM=.25;
T_ClBlock=Time_CL/60;
T_D=0;
T_AM=0.1;
T_Cruise=((D_Block+.02*D_Block+20)-(Range_CL))/(V_Cr*1.15);
% Velocity in MPH
V_Block=D_Block/(T_GM+T_ClBlock+T_D+T_Cruise+T_AM);

% Block Time
T_Block= T_GM+T_ClBlock+T_D+T_Cruise+T_AM;

% Block Fuel
F_ClBlock=W_FuelCL;
F_CrAM=T_RR*(c_Cruise*ADV_SFC*(T_Cruise+T_AM));
F_Block=F_ClBlock+F_CrAM;

%% Flying Operating Cost

% Flight Crew
PL = W_PL/2000; % Convert to tons
V_Crnm= V_Cr*1.15;
if NFCrew == 2
    Dol_BH=17.849*(V_Crnm*(W_TO/10^5))^.3+40.83;
elseif NFCrew == 3
    Dol_BH=24.261*(V_Crnm*(W_TO/10^5))^.3+57.62;
```

```
end
C_TMFC=Dol_BH/(V_Block*PL);

% Fuel & Oil

C_Fuel=.0625;
C_Oil=2.15;
C_TMFO=(1.02*F_Block*C_Fuel+num_eng*C_Oil*T_Block*.135)/(D_Block*PL);

% Hull Insurance

W_A=W_TO-(W_Fuel*W_TO)-W_PL-(W_PP*W_TO);
C_Airframe=(2.4*10^6)+(87.5*W_A);
C_Engine=(590000+16*TC)*ADV_Eng;
C_ATotal=C_Airframe+(num_eng*C_Engine);

IRA=0.1;
U=630+4000/(1+(1/T_Block+.5));
C_TMHI=(IRA*C_ATotal)/(U*V_Block*PL);

%% Direct Maintenance

% Airframe Labor

K_FHA=4.9169*log10(W_A/(10^3))-6.425;
K_FCA=.21256*log10(W_A/(10^3))^3.7375;
T_MF=T_Block-T_GM;
R_Labor=8.60;
C_TMAL=(K_FHA*T_MF+K_FCA)/(V_Block*T_Block*PL)*R_Labor;

% Airframe Material

C_FHA=(1.5994*C_Airframe/10^6)+3.4263;
C_FCA=(1.9229*C_Airframe/10^6)+2.2504;
C_TMAM=(C_FHA*T_MF+C_FCA)/(V_Block*T_Block*PL);

% Engine Labor

K_FHE=(num_eng*TC/10^3)/(.82715*(TC/10^3)+13.639);
K_FCE=.20*num_eng;
C_TMEL=(K_FHE*T_MF+K_FCE)/(V_Block*T_Block*PL)*R_Labor;

% Engine Material

C_FHE=(28.2353*C_Engine/10^6-6.5176)*num_eng;
C_FCE=(3.6698*C_Engine/10^6+1.3685)*num_eng;

C_TMEM=(C_FHE*T_MF+C_FCE)/(V_Block*T_Block*PL);

% Total Maintence

C_MTotal=C_TMAL+C_TMAM+C_TMEL+C_TMEM*2;
```

```
%% Depreciation

C_MDep=(1/(V_Block*PL))*(C_ATotal+.06*(C_ATotal-
(num_eng*C_Engine))+.3*num_eng*C_Engine)/(14*U);

%% Total DOC

Total_DOC=C_TMFC+C_TMFO+C_TMHI+C_MTotal+C_MDep % Per Ton-mile

Pass_Mile= Total_DOC*PL/Pas
```