**TUTORIAL 1:**

Matlab code-

answer1 = inputdlg('Enter abscissa of vertices of the Triangular membership function A in order: ', 'INPUT A', [1 50]);

val1 = str2num(answer1{1});

N = 16001;

minx = -20;

maxx = 20;

x = linspace(minx,maxx,N);

y1 = trimf(x, val1);

answer2 = inputdlg('Enter abscissa of vertices of the Triangular membership function B in order: ', 'INPUT B', [1 50]);

val2 = str2num(answer2{1});

y2 = trimf(x, val2);

inv = [1/val2(3), 1/val2(2), 1/val2(1)];

sum = [val1(1)+val2(1), val1(2)+val2(2), val1(3)+val2(3)];

diff = [val1(1)-val2(3), val1(2)-val2(2), val1(3)-val2(1)];

y = trimf(x, inv);

y3 = trimf(x, sum);

y4 = trimf(x, diff);

y5 = fuzarith(x,y1,y2,'prod');

y6 = fuzarith(x,y1,y,'prod');

figure

subplot(4,1,1)

plot(x,y1,'b--',x,y2,'m:',x,y3,'c')

title('Fuzzy Addition, A+B')

legend('A','B','A+B')

subplot(4,1,2)

plot(x,y1,'b--',x,y2,'m:',x,y4,'c')

title('Fuzzy Subtraction, A-B')

legend('A','B','A-B')

subplot(4,1,3)

plot(x,y1,'b--',x,y2,'m:',x,y5,'c')

title('Fuzzy Product, A\*B')

legend('A','B','A\*B')

subplot(4,1,4)

plot(x,y1,'b--',x,y2,'m:',x,y6,'c')

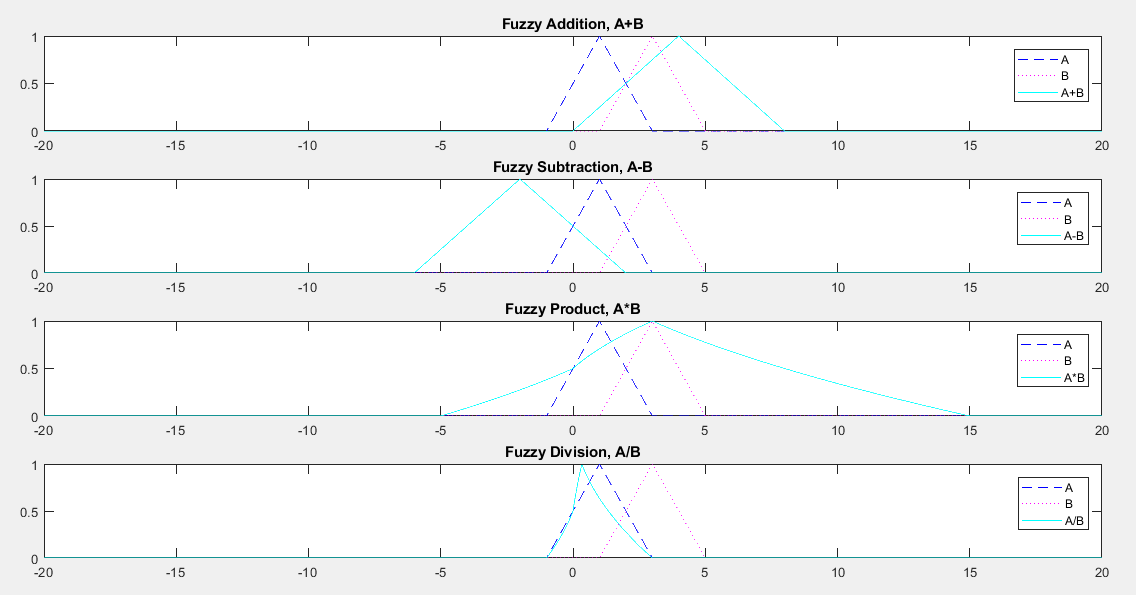
title('Fuzzy Division, A/B')

legend('A','B','A/B')

Input : A = [-1 1 3]

B = [1 3 5]

Graph obtained:



**TUTORIAL 2:**

Matlab Code-

N = 2001;

minx = 0;

maxx = 1.25;

x = linspace(minx,maxx,N);

P\_zero=trimf(x, [0 0 0.5]);

P\_low=trimf(x, [0 0.5 1]);

P\_high=trimf(x, [0.5 1 1]);

T\_zero=trimf(x, [0 0 0.25]);

T\_low=trimf(x, [0 0.25 0.5]);

T\_high=trimf(x, [0.25 1 1]);

FR\_zero=trimf(x, [0 0 0.125]);

FR\_low=trimf(x, [0 0.125 0.25]);

FR\_high=trimf(x, [0.125 1 1]);

ans = inputdlg({'Pressure(enter crisp): ','Temperature(enter crisp): ','Flow Rate(enter crisp): '}, 'Crisp Input', [1 30; 1 30; 1 30]);

for r=1:3

val(r) = str2num(ans{r});

end

P=P\_high(val(1)\*(2000/8)+1);

T=T\_high(val(2)\*(2000/800)+1);

FR=FR\_zero(val(3)\*(2000/80)+1);

Auto=0.5\*P+0.25\*T+0.25\*FR;

m=Auto;

P=P\_high(val(1)\*250+1);

T=T\_low(val(2)\*2.5+1);

FR=FR\_zero(val(3)\*25+1);

Ann=0.5\*P+0.25\*T+0.25\*FR;

if Ann>m

m=Ann;

max='Annealing';

end

P=P\_low(val(1)\*250+1);

T=T\_zero(val(2)\*2.5+1);

FR=FR\_low(val(3)\*25+1);

Sin=0.5\*P+0.25\*T+0.25\*FR;

if Sin>m

m=Sin;

max='Sintering';

end

P=P\_zero(val(1)\*250+1);

T=T\_zero(val(2)\*2.5+1);

FR=FR\_high(val(3)\*25+1);

Trans=0.5\*P+0.25\*T+0.25\*FR;

if Trans>m

m=Trans;

max='Transport';

end

ans1 = inputdlg({'Pressure(enter abscissa of triangular membership function in the format- a(left) b(center) c(right)): ','Temperature(enter abscissa of triangular membership function in the format- a(left) b(center) c(right)): ','Flow Rate(enter abscissa of triangular membership function in the format- a(left) b(center) c(right)): '}, 'Fuzzy Input', [1 150; 1 150; 1 150]);

val1 = str2num(ans1{1})/8;

val2 = str2num(ans1{2})/800;

val3 = str2num(ans1{3})/80;

auto=[0.5 1 1]\*0.5+[0.25 1 1]\*0.25+[0 0 0.125]\*0.25;

ac=trimf(x, auto);

c1=(ac(1)+ac(2)+ac(3))/3;

ann=[0.5 1 1]\*0.5+[0 0.125 0.5]\*0.25+[0 0 0.125]\*0.25;

an=trimf(x, ann);

c2=(an(1)+an(2)+an(3))/3;

sin=[0 0.5 1]\*0.5+[0 0 0.125]\*0.25+[0 0.125 0.25]\*0.25;

s=trimf(x, sin);

c3=(s(1)+s(2)+s(3))/3;

trans=[0 0 0.5]\*0.5+[0 0 0.25]\*0.25+[0.125 1 1]\*0.25;

t=trimf(x, trans);

c4=(t(1)+t(2)+t(3))/3;

input=val1\*0.5+val2\*0.25+val3\*0.25;

in=trimf(x, input);

ci=(in(1)+in(2)+in(3))/3;

figure

subplot(4,1,1)

plot(x,P\_zero,'b--',x,P\_low,'m:',x,P\_high,'c')

title('Pressure(scaled 8 kPa)')

legend('Zero','Low','High')

subplot(4,1,2)

plot(x,T\_zero,'b--',x,T\_low,'m:',x,T\_high,'c')

title('Temperature(scaled 800 deg. Celsius)')

legend('Zero','Low','High')

subplot(4,1,3)

plot(x,FR\_zero,'b--',x,FR\_low,'m:',x,FR\_high,'c')

title('Flow Rate(scaled 80 gph)')

legend('Zero','Low','High')

subplot(4,1,4)

plot(x,ac,'b--',x,an,'m:',x,s,'c',x,t,':',x,in,'g--')

legend('Autoclaving','Annealing','Sintering','Transport','INPUT')

min=abs(ci-c1);

m='Autoclaving';

if abs(ci-c2)<=min

m='Annealing';

min=abs(ci-c2);

end

if abs(ci-c3)<=min

m='Sintering';

min=abs(ci-c3);

end

if abs(ci-c4)<=min

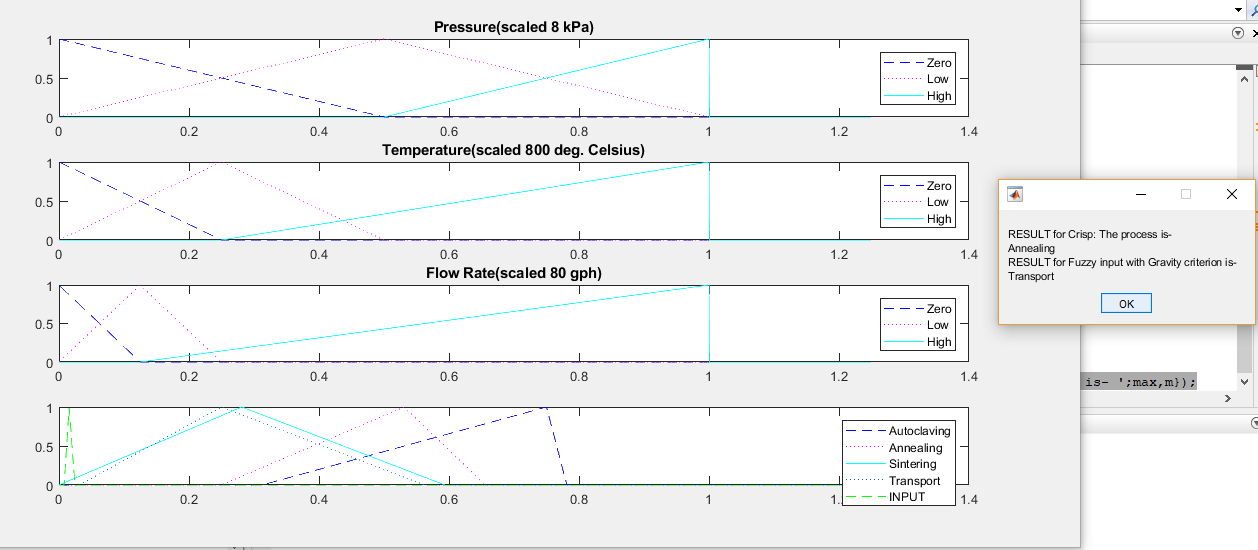
m='Transport';

end

f = msgbox({'RESULT for Crisp: The process is- ','RESULT for Fuzzy input with Gravity criterion is- ';max,m});

Input- Crisp P=5kPa, T=150 degree C, FR=5gph

Fuzzy P=[ 1/8 1/4 3/8 ], T=[1/8 3/8 5/8], FR=[0 0 1/4]

Graph(for fuzzy) and Result Message Prompt(for crisp)

**TUTORIAL 3:**

Matlab Code-

answer=inputdlg({'Enter the interval 1 in the format: a b ','Enter the interval 2: ','Enter the interval 3: ','Enter the interval 4: '},'INTERVALS',[1 50;1 50;1 50;1 50]);

A = str2num(answer{1});

B = str2num(answer{2});

C = str2num(answer{3});

D = str2num(answer{4});

sets = {A, B, C, D};

[w x y z] = ndgrid(sets{:});

I = [w(:) x(:) y(:) z(:)];

ans=inputdlg({'Enter the fuzzy weight 1 in the format: w1 w2 ','Enter the fuzzy weight 2: ','Enter the fuzzy weight 3: ','Enter the fuzzy weight 4: '},'WEIGHTS',[1 50;1 50;1 50;1 50]);

W1 = str2num(ans{1});

W2 = str2num(ans{2});

W3 = str2num(ans{3});

W4 = str2num(ans{4});

sets = {W1, W2, W3, W4};

[w x y z] = ndgrid(sets{:});

W = [w(:) x(:) y(:) z(:)];

Y=zeros(256,1);

t=1;

for r=1:16

for c=1:16

V = dot(I(r,:),W(c,:),2);

S = sum(W(c,:),2);

Y(t,1) = V./S;

t=t+1;

end

end

m=min(Y);

M = max(Y);

y = [m M];

disp('The Interval of Y is: ');

disp(y);

Input - as given in class for the fuzzy-weights with fuzzy intervals problem

Output – matches with that given in class

**TUTORIAL 4:**

Matlab Code-

rng('shuffle')

for r=1:20

X(r,:)=[round(rand(1,10)) round(rand(1,10))];

end

C=zeros(20,1);

f=zeros(20,1);

F=zeros(20,1);

A=zeros(20,1);

B=zeros(20,1);

C=zeros(20,1);

D=zeros(20,1);

E=zeros(20,1);

for s=1:60

for r=1:20

x=6\*bi2de(X(r,1:10))/1023;

y=6\*bi2de(X(r,11:20))/1023;

f(r,1) = (x\*x+y-11)\*(x\*x+y-11)+(x+y\*y-7)\*(x+y\*y-7);

F(r,1)=1/(1+f(r,1));

Tot=sum(F);

A(r,1)=20\*F(r,1)/Tot;

B(r,1)=A(r,1)/20;

C(r,1)= sum(B(1:r,1));

D(r,1)=rand;

end

n=20;

for r=1:20

while D(r,1)<C(n,1) && n>1

n = n-1;

end

if n==20

E(r,1)=n;

else E(r,1)=n+1;

end

S(r,1:10)=X(E(r,1),1:10);

S(r,11:20)=X(E(r,1),11:20);

n=20;

end

for r=1:10

o=randperm(20);

o2=randperm(10);

i = o(1);

j = o(2);

k = o2(1);

t=S(i,k:10);

S(i,k:10)=S(j,k:10);

S(j,k:10)=t;

t=S(i,(10+k):20);

S(i,(10+k):20)=S(j,(10+k):20);

S(j,(10+k):20)=t;

X(i,1:20)=S(i,1:20);

X(j,:)=S(j,:);

end

[m,I]=max(F);

disp(F);

if abs(m-1)<0.1

x=6\*bi2de(X(I,1:10))/1023;

y=6\*bi2de(x(I,11:20))/1023;

disp('The value close to solution is: ');

disp([x y]);

break

end

end

Runs the GA iterations for problem given in class.