**Practical No : - 4**

**Aim : Write a program to implement fuzzy operation on relation set.**

**Code :**

#include<stdio.h>

#include<conio.h>

#include "Header.H"

void main()

{

int i,j,k,ch;

float a[3][3]={0.1,0.2,0.7,0.4,0.5,0.73,0.7,0.11,0.9},b[3][3]={0.7,0.12,0.3,0.41,0.5,0.3,0.70,0.8,0.9},max,min,ans[3][3],temp,bi1;

clrscr();

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("\t%.2f",a[i][j]);

}

printf("\n");

}

printf("\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("\t%.2f",b[i][j]);

}

printf("\n");

}

printf("\n 1.Union 2.Intersection 3.Composition max-min 4.Composition max-star");

printf("\n Enter the Value:");

scanf("%d",&ch);

printf("\n\n");

switch(ch)

{

case 1: {

printf("\n\tUnion of Matrix :\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

ans[i][j]=uni(a[i][j],b[i][j]);

printf("\t%.2f",ans[i][j]);

}

printf("\n");

}

break;

}

case 2:{

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

ans[i][j]=inter(a[i][j],b[i][j]);

printf("\t%.2f",ans[i][j]);

}

printf("\n");

}

break;

}

case 3:{

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

max=-1;

for(k=0;k<3;k++)

{

max=uni(max,inter(a[i][k],b[k][j]));

}

ans[i][j]=max; }

}

}

printf("\nMAX MIN of a & b is");

printf("\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("\t%.2f",ans[i][j]);

}

printf("\n");

}

break;

case 4: {

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

max=-1;

for(k=0;k<3;k++)

{

max=uni(max,(a[i][k]\*b[k][j]));

}

ans[i][j]=max; }

}

}

printf("\nMAX MIN of a & b is");

printf("\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

printf("\t%.2f",ans[i][j]);

}

printf("\n");

}

break;

}

getch();

}

**Header File :**

float uni(float p,float q)

{

if(p>=q)

{

return p;

}

else

{

return q;

}

}

float inter(float w,float x)

{

if(w>=x)

{

return x;

}

else

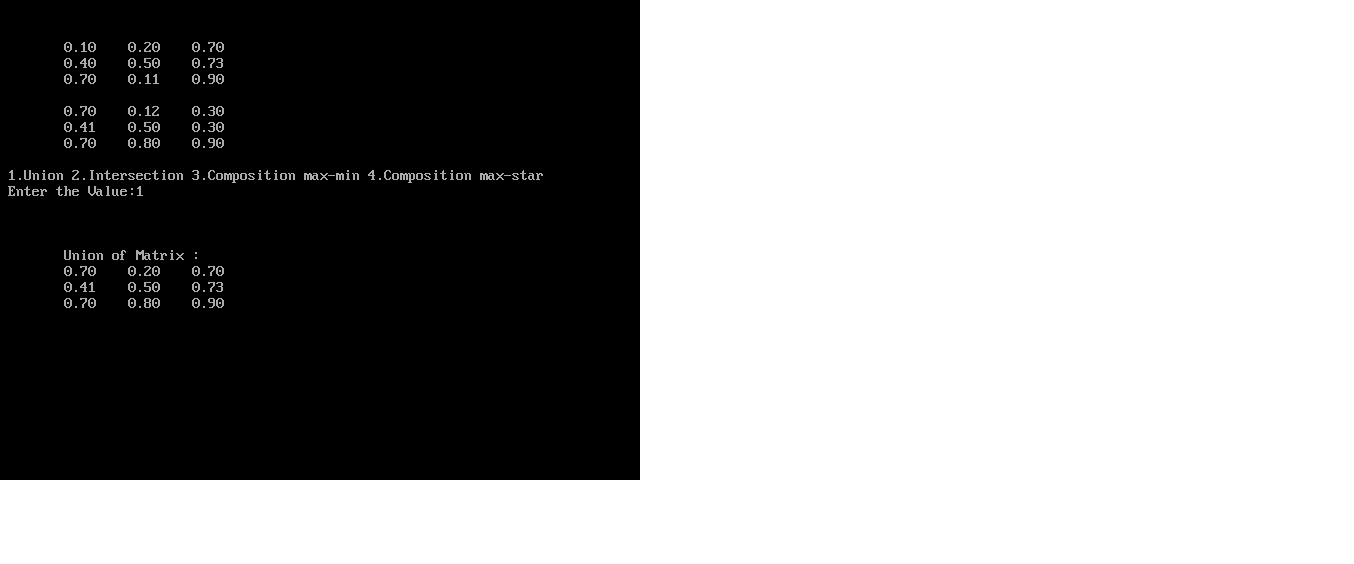
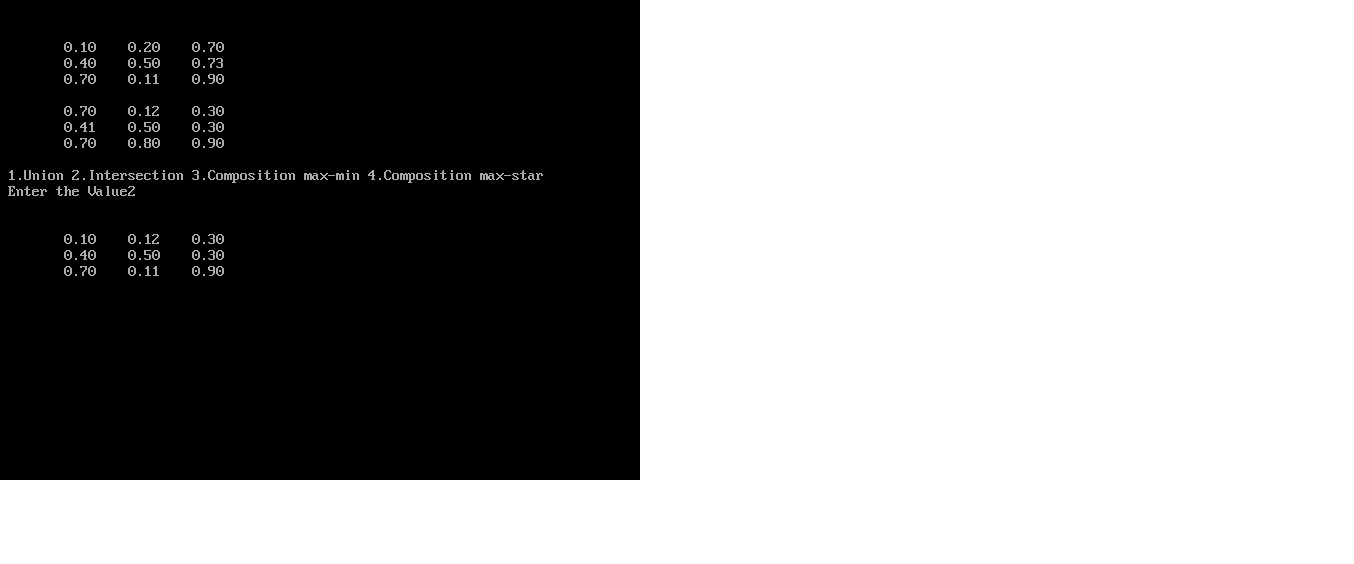
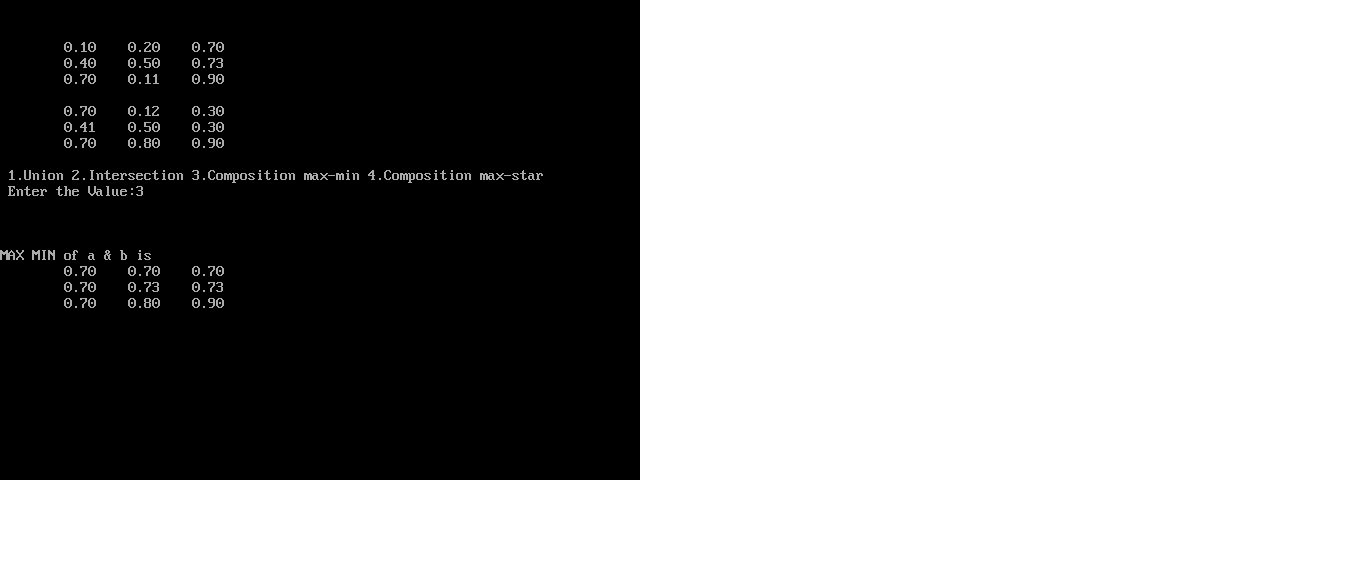
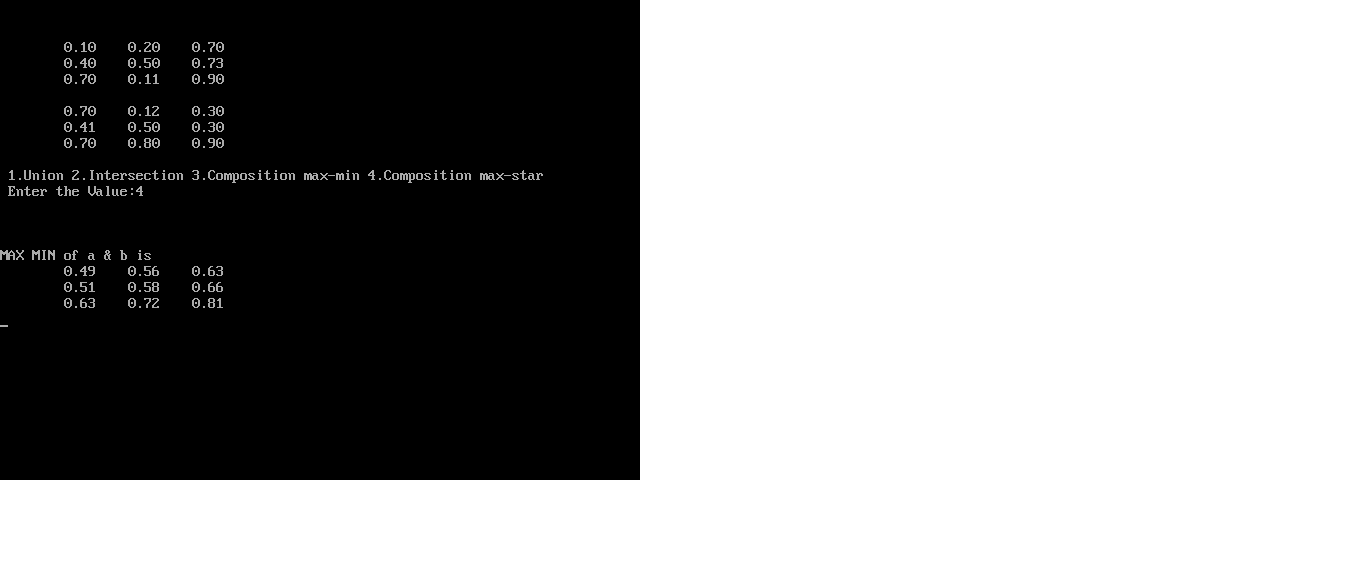
{

return w;

}

}

**Output:**

****