```
#include<stdio.h>
#include<stdlib.h>

typedef struct SET
{
      float nr[100];
        float dr[100];
      int n;
}fuzzy;

//create fuzzy set

fuzzy *getval(fuzzy *m,int ch,char *x)
{
int i;
float f,g;
m=(fuzzy*)malloc(sizeof(fuzzy*));
m->n=ch;
printf("Enter elements in %s\n",x);
for(i=0;i<m->n;i++)
{
printf("Enter the value of numerator %d\n",i+1);
scanf("%f",&f);
m->nr[i]=f;
printf("Enter the value of demorator %d\n",i+1);
scanf("%f",&g);
m->dr[i]=g;
}
return(m);
}

void printval(fuzzy *m,char *x)
{
int i;
printf("\n the set %s {",x);
for(i=0;i<m->n;i++)
{
printf("%6.1f / %.0f",m->nr[i],m->dr[i]);
if(i!=m->n-1)
printf(",");
}
printf("}");
}
// union of two sets
fuzzy *unionset(fuzzy *a,fuzzy *b)
{
fuzzy *temp=NULL;
temp=(fuzzy *)malloc(sizeof(fuzzy *));
int i;
temp->n=a->n;
for(i=0;i<a->n;i++)
{
      if(a->dr[i]==b->dr[i])
          {
```

```
          if(a->nr[i]>=b->nr[i])
           {
           temp->dr[i]=a->dr[i];
             temp->nr[i]=a->nr[i];
            }
         else
      {
            temp->dr[i]=b->dr[i];
             temp->nr[i]=b->nr[i];
      }
       }
 }
return(temp);
}

void main()
{
fuzzy *a=NULL;
fuzzy *b=NULL;
fuzzy *result=NULL;
int c,ch;
printf("How many components in sets\n");
scanf("%d",&c);
a=getval(a,c,"A");
printval(a,"A");
b=getval(b,c,"B");
printval(b,"B");
do
{
printf("1.UNION\n2.INTERSECTION\n3.COMPLEMENT\n4.EXIT\n");
printf("Enter your choice\n");
scanf("%d",&ch);
switch(ch)
{
case 1:result=unionset(a,b);
       printval(result,"C");
       break;
/*case 2:result=intersection(a,b);
       printval(result,"D");
       break;
case 3:result=complement(a,b);
       printval(result,"E");
       break;*/
case 4:exit;
       break;
default:printf("InValid choice");
       break;
}
}while(ch!=4);

}
```