Lab 1

1. **Write a program that takes two or more sets as input and produces set operations like union, intersection, difference and symmetric difference as its output.**

#include<iostream>

using namespace std;

int main()

{

int a[10],b[10],c[10],fl=0;

int i,j,n1,n2,k=0,x=0;

cout<<"How many element are there in set A: ";

cin>>n1;

cout<<"Enter elements of set A: ";

for(i=0;i<n1;i++)

{

cin>>a[i];

}

cout<<"\nHow many element are there in set B: ";

cin>>n2;

cout<<"Enter elements of set B: ";

for(j=0;j<n2;j++)

{

cin>>b[j];

}

cout<<endl<<"A: { ";

for(i=0;i<n1;i++)

{

cout<<a[i]<<" ";

}

cout<<"}"<<endl;

cout<<"B: { ";

for(j=0;j<n2;j++)

{

cout<<b[j]<<" ";

}

cout<<"}"<<endl<<endl;

// Union-------------

{

k=0;

for(i=0;i<n1;i++)

{

c[x]=a[i];

x++;

}

k=x;

for(i=0;i<n2;i++)

{

fl=0;

for(j=0;j<n1;j++)

{

if(a[j]==b[i])

{

fl=0;

break;

}

else

fl=1;

}

if(fl==1)

{

c[k]=b[i];

k++;

}

}

cout<<endl<<"Union: { ";

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

{

cout<<c[i]<<" ";

}

cout<<"}";

}

// Intersection-----------

k=0,x=0;

for(i=0;i<n1;i++)

{

for(j=0;j<n2;j++)

{

if(a[i]==b[j])

{

c[k]=a[i];

k++;

}

}

}

cout<<endl<<"Intersection: { ";

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

{

cout<<c[i]<<" ";

}

cout<<"}";

// A-B-------------------------

k=0;

for(i=0;i<n1;i++)

{

fl=0;

for(j=0;j<n2;j++)

{

if(a[i]==b[j])

{

fl=0;

break;

}

else

{

fl=1;

}

}

if(fl==1)

{

c[k]=a[i];

k++;

}

}

cout<<endl<<"Diff. of A - B: { ";

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

{

cout<<c[i]<<" ";

}

cout<<"}"<<endl;

// B-A---------------------------

k=0;

for(i=0;i<n2;i++)

{

fl=0;

for(j=0;j<n1;j++)

{

if(b[i]==a[j])

{

fl=0;

break;

}

else

fl=1;

}

if(fl==1)

{

c[k]=b[i];

k++;

}

}

cout<<"Diff. of B - A: { ";

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

{

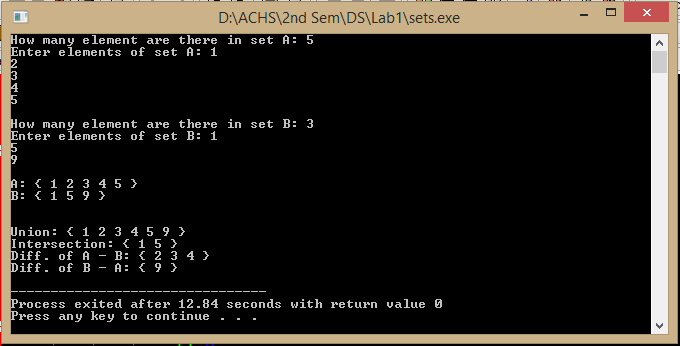
cout<<c[i]<<" ";

}

cout<<"}"<<endl;

}

**Output:**



1. **Write a program that takes two or more sets as input and produces their Cartesian product as output.**

#include<iostream>

using namespace std;

int main()

{

int a[10],b[10];

int i,j,n1,n2;

cout<<"How many element are there in set A: ";

cin>>n1;

cout<<"Enter elements of set A: ";

for(i=0;i<n1;i++)

{

cin>>a[i];

}

cout<<"\nHow many element are there in set B: ";

cin>>n2;

cout<<"Enter elements of set B: ";

for(j=0;j<n2;j++)

{

cin>>b[j];

}

cout<<endl<<"A: { ";

for(i=0;i<n1;i++)

{

cout<<a[i]<<" ";

}

cout<<"}"<<endl;

cout<<"B: { ";

for(j=0;j<n2;j++)

{

cout<<b[j]<<" ";

}

cout<<"}"<<endl<<endl;

cout<<"Cartesian product: ";

for(i=0;i<n1;i++)

{

for(j=0;j<n2;j++)

{

cout<<"{"<<a[i]<<","<<b[j]<<"}";

cout<<" ";

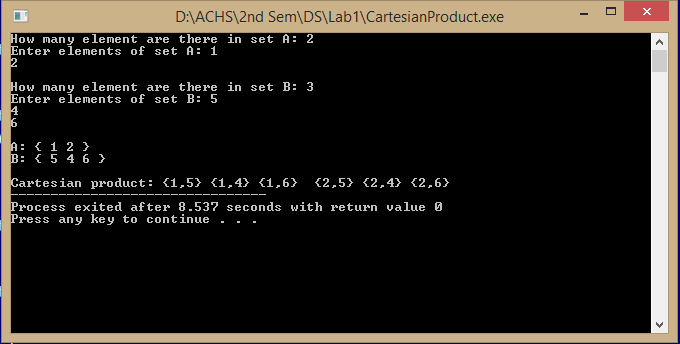
}

cout<<" ";

}

}

**Output:**



1. **Write a program that takes a real number and produces is ceiling and floor integers as output.**

#include<iostream>

using namespace std;

int main()

{

int num;

float x;

cout<<"Enter a value: ";

cin>>x;

cout<<"Floor: " ;

if(x==int(x))

{

cout<<x;

}

else{

if(x<0)

{

cout<<int(x)-1;

}

else{

cout<<int(x);

}

}

cout<<endl<<"Ceiling: " ;

if(x<=0)

{

cout<<int(x);

}

else

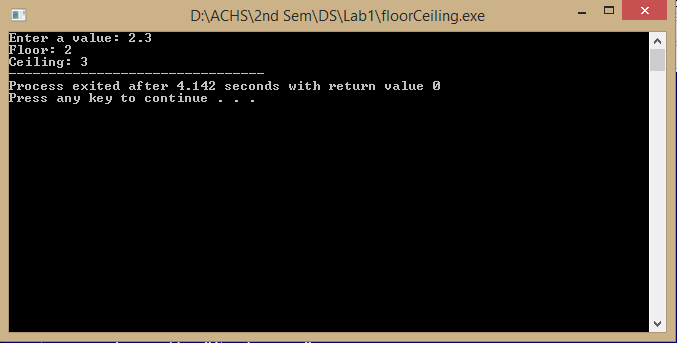
{

cout<<int(x)+1;

}

}

**Output:**



1. Write a program that takes name and age of a 5 persons as an input and gives the degree of membership of the person as its output according to following membership functions.

a. Degree of membership = 1 if age<=20

Degree of membership = (30-age)/10 if age>20 and age<=30

Degree of membership = 0 if age>30

b. Degree of membership = 1 if age<=15

Degree of membership = (35-age)/20 if age>15 and age<=35

Degree of membership = 0 if age>35

Perform set operations according to rules of fuzzy sets, on these

two sets.

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<iostream>

using namespace std;

void create1();

void create2();

void membership1();

void membership2();

struct student

{

int age;

char fname[20];

float degree,send[10];

}st;

FILE \*fptr;

int main()

{

int ch;

do

{

system("cls");

cout<<"\t=======================\n\tMAIN MENU\n\t=======================\n";

cout<<"\t[1] Question Number (a): \n";

cout<<"\t[2] Question Number (b):\n";

cout<<"\t[3] See Membership 1:\n";

cout<<"\t[4] See Membership 2:\n";

cout<<"\t[5] EXIT\n\t=======================\n\n";

cout<<"\n\tPlease Select From Above Option: ";

cin>>ch;

switch(ch)

{

case 1:

create1();

break;

case 2:

create2();

break;

case 3:

membership1();

break;

case 4:

membership2();

break;

case 5:

exit(0);

default :

cout<<"\nInvalid Choice!!!\a";

}

}while(ch!=5);

}

void create1()

{

int i;

system("cls");

cout<<"===================================\nPLEASE ENTER NEW DETAILS OF STUDENT\n===================================\n";

for(i=0;i<5;i++){

fptr=fopen("membership1.dat","ab");

cout<<"Enter First Name of student: ";

cin>>st.fname;

cout<<"Enter age of the student: ";

cin>>st.age;

cout<<endl<<endl;

if(st.age>30)

st.degree=1;

else if(st.age>20 && st.age<=30)

st.degree=(30-st.age)/10;

else

st.degree=0;

fwrite(&st,sizeof(st),1,fptr);

fclose(fptr);

}

cout<<endl<<"New Student Record Added\nPress Any Key To Continue...\n";

getch();

}

void create2()

{

int i;

system("cls");

cout<<"===================================\nPLEASE ENTER NEW DETAILS OF STUDENT\n===================================\n";

for(i=0;i<5;i++){

fptr=fopen("membership2.dat","ab");

cout<<"Enter First Name of student: ";

cin>>st.fname;

cout<<"Enter age of the student: ";

cin>>st.age;

cout<<endl<<endl;

if(st.age>35)

st.degree=0;

else if(st.age>15 && st.age<=35)

st.degree=(35-st.age)/20;

else

st.degree=1;

fwrite(&st,sizeof(st),1,fptr);

fclose(fptr);

}

cout<<endl<<"New Student Record Added\nPress Any Key To Continue...\n";

getch();

}

void membership1()

{

system("cls");

cout<<"\n\n\n\t\tDISPLAY RECORDS OF GROUP A!!!\n\n";

fptr=fopen("membership1.dat","rb");

while((fread(&st,sizeof(st),1,fptr))>0)

{

cout<<"Name: "<<st.fname<<endl;

cout<<"Age: "<<st.age<<endl;

cout<<"Degree of membership: "<<st.degree<<endl;

cout<<"\n\n============================\n";

}

fclose(fptr);

getch();

}

void membership2()

{

system("cls");

cout<<"\n\n\n\t\tDISPLAY RECORDS OF GROUP B!!!\n\n";

fptr=fopen("membership2.dat","rb");

while((fread(&st,sizeof(st),1,fptr))>0)

{

cout<<"Name: "<<st.fname<<endl;

cout<<"Age: "<<st.age<<endl;

cout<<"Degree of membership: "<<st.degree<<endl;

cout<<"\n\n============================\n";

}

fclose(fptr);

getch();

}

**Output:**

