**ASSIGNMENT 1**

**Aim** -Develop an object oriented program in c++ to create a student database of system containing the following information

1]Name 2]Roll\_no 3]Class 4]Division 5]DOB 5]Blood group

6]Address 7]Telephone 8]Driving license no.

with functions static member functions,friend function and also pointer,Dynamic memory allocation

**Objective**-Forming a structure,calling friend function and as well as giving dynamic memory allocation defining class application of all the above things

**Theory**-

Friend Function Class A friend class can access private and protected members of other class in which it is declared as friend.

The concept of dynamic memory allocation in c language enables the C programmer to allocate memory at runtime. Dynamic memory allocation in c language is possible by 4 functions of stdlib.h header file.

malloc() calloc() realloc()

A static member function is a special member function, which is used to access only static data members, any other normal data member cannot be accessed through static member function.

**Algorithm**-

Step1 :start

Step2:create class of students database with name of student,roll number of student,division,class,address,contact number of students.

Step3:Using friend function take the input value from users and display of database.

Step4:Give the option to user to add new data,delete data and modify data.

Step5:Stop.

**Source code-**

#include<iostream>

#include<string.h>

using namespace std;

class data

{

private:

char name[30];

int roll;

char div;

char cls[30],blood[10],contact[20];

char lic[30],adress[30];

public:

void input()

{

cout<<"Enter name of student :";

cin>>name;

cout<<"Enter roll no of student :";

cin>>roll;

cout<<"Enter divison of student :";

cin>>div;

cout<<"Enter class of student :";

cin>>cls;

cout<<"Enter blood group of student :";

cin>>blood;

cout<<"Enter contact of student :";

cin>>contact;

cout<<"Enter license of student :";

cin>>lic;

cout<<"Enter adress of student :";

cin>>adress;

}

void edit(int a2,int n);

friend void profile(data);

friend void delet(data,data,int a3,int n);

};

void profile(data b)

{

cout<<endl<<b.name<<" "<<b.roll<<" "<<b.div<<" "<<b.cls<<" "<<b.blood<<" "<<b.contact<<" "<<b.lic<<" "<<b.adress;

}

void data::edit(int a2,int n)

{

if(roll==a2)

{

input();

}

}

void delet(data b,data d,int a3,int n)

{

if(b.roll==a3)

{

b.roll=d.roll;

b.div=d.div;

b.contact=d.contact;

strcpy(b.name,d.name);

strcpy(b.cls,d.cls);

strcpy(b.blood,d.blood);

strcpy(b.lic,d.lic);

strcpy(b.adress,d.adress);

}

}

int main()

{

int n,i,j,a,a1,a2,a3;

cout<<"Enter the number of students:";

cin>>n;

data b[100];

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

{

b[j].input();

}

cout<<"NAME "<<"ROLL "<<"DIV "<<"CLASS "<<"BLOOD "<<"CONTACT "<<"DRI. LICENCE "<<"ADRESS "<<endl;

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

{

profile(b[j]);

}

cout<<endl<<"1.Add"<<endl<<"2.Delete"<<endl<<"3.Modify"<<endl;

cin>>a;

switch(a)

{

case 1:

cout<<"Enter the number of students:";

cin>>a1;

for(i=n;i<(a1+n);i++)

{

b[i].input();

}

for(j=0;j<(a1+n);j++)

{

profile(b[j]);

}

break;

case 2:

cout<<"Which roll number you want to delete:";

cin>>a3;

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

{

delet(b[i],b[i+1],a3,n);

}

for(j=0;j<(n-1);j++)

{

profile(b[j]);

}

break;

case 3:

cout<<"Which roll number you want to edit :";

cin>>a2;

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

{

b[i].edit(a2,n);

}

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

{

profile(b[j]);

}

break;

}

return 0;

}

**Output**-

Enter name of student :abhi

Enter roll no of student :1

Enter divison of student :d

Enter class of student :sy

Enter blood group of student :o

Enter contact of student :123

Enter license of student :abc123

Enter adress of student :pune

NAME ROLL DIV CLASS BLOOD CONTACT DRI. LICENCE ADRESS

abhi 1 d sy o 123 abc123 pune

1.Add

2.Delete

3.Modify

1

Enter the number of students:1

Enter name of student :aniket

Enter roll no of student :2

Enter divison of student :d

Enter class of student :sy

Enter blood group of student :ab

Enter contact of student :456

Enter license of student :asd456

Enter adress of student :satara

abhi 1 d sy o 123 abc123 pune

aniket 2 d sy ab 456 asd456 satara

**Conclusion**-

Learning all the above points of object oriented application and staatic member function as well as dynamic memory allocation and alse mainly the use of friend function