ASSIGNMENT 8

AIM:-

Department maintains a student information.the file contains roll number,name,division and address .Allow user to add,delete information of student.display information of particular employee.If record of student does not exists an appropriate message is displayed.If it is, then the system displays,the student details.use sequential file to main the data.

OBJECTIVE:-

To implement file handling and perform functions like insertion,deletion and display of record using sequential file.

THEORY:-

A **sequential file** is one that contains and stores data in chronological order. The data itself may be ordered or un ordered in the file. Unlike a random-access file, sequential files must be read from the beginning, up to the location of the desired data. Sequential files are often stored on sequential access devices, like a magnetic tape.

A sequential file contains records organized by the order in which they were entered. The order of the records is fixed.
Records in sequential files can be read or written only sequentially.
After you place a record into a sequential file, you cannot shorten, lengthen, or delete the record. However, you can update a record if the length does not change. New records are added at the end of the file.
If the order in which you keep records in a file is not important, sequential organization is a good choice whether there are many records or only a few. Sequential output is also useful for printing reports.

ALGORITHM:-

1.CREATE A FILE HAVING COLLECTION OF RECORDS

```
void Create()
{
 char ch='y';
 ofstream seqfile;
 seqfile.open("stud.DAT",ios::out|ios::binary);
 do
 {
```

```
  cout<<"\n Enter roll no: ";
  cin>>Records.rollno;
  seqfile.write((char*)&Records,sizeof(Records));
  cout<<"\nDo you want to add more records?";
  cin>>ch;
  }while(ch=='y');
  seqfile.close();
}
```

2.DISPLAY OF FILE

```
Void Display()
{
 ifstream seqfile;
 seqfile.open("stud.DAT",ios::in|ios::binary);
 seqfile.seekg(0,ios::beg);
 cout<<"\n The Contents of file are ..."<<endl;
 while(seqfile.read((char *)&Records,sizeof(Records)))
 {
   if(Records.rollno!=-1)
   {
   cout<<"\nRoll No: "<<Records.rollno;
    }
 }
 seqfile.close();
}
```

3.SEARCHING A RECORD

```
int :Search()
{
 fstream seqfile;
 int id,pos,offset;
 cout<<"\n Enter the no for searching the record ";
 cin>>id;
 seqfile.open("stud.DAT",ios::in|ios::binary);
```

```
 pos=-1;
 seqfile.seekg(0,ios::beg);
 int i=0;
 while(seqfile.read((char *)&Records,sizeof(Records)))
 {
  if(id==Records.rollno)
  {
   pos=i;
   break;
  }
  i++;
 }
  seqfile.close();
  return pos;
}
```

4.DELETION OF RECORD:-

```
void deletion()
{
    int id,pos;
    cout<<"For deletion"<<endl;
    fstream seqfile;
    pos=Search();
    seqfile.open("stud.DAT",ios::in|ios::binary|ios::out);
    seqfile.seekg(0,ios::beg);
    if(pos==-1)
    {
    cout<<"\n Record is not present in the file";
    return;
    }
  int offset=pos*sizeof(Records);
  seqfile.seekp(offset);
  Records.rollno=-1;
```

```
  seqfile.write((char *)&Records,sizeof(Records));
  seqfile.seekg(0);
  seqfile.close();

}
```

#endif // SDASSIGNMENT8_CPP_INCLUDED

PROGRAM CODE:-

```
#include<iostream>
#include<fstream>
#include<string.h>
using namespace std;
typedef struct data
 {
 char name[10];
 int rollno;
 char div;
 char address[100];
 }Rec;
 class student
{

 Rec Records;
 public:
  void Create();
  void Display();
  int Search();
  void deletion();
};
```

```
void student::Create()
{
 char ch='y';
 ofstream seqfile;
 seqfile.open("stud.DAT",ios::out|ios::binary);
 do
 {
  cout<<"\n Enter Name: ";
  cin>>Records.name;
  cout<<"\n Enter roll no: ";
  cin>>Records.rollno;
  cout<<"\n Enter division";
  cin>>Records.div;
  cout<<"\n Enter ADDRESS: ";
  cin>>Records.address;
  seqfile.write((char*)&Records,sizeof(Records));
  cout<<"\nDo you want to add more records?";
  cin>>ch;
  }while(ch=='y');
  seqfile.close();
}
void student::Display()
{
 ifstream seqfile;
 seqfile.open("stud.DAT",ios::in|ios::binary);
 seqfile.seekg(0,ios::beg);
 cout<<"\n The Contents of file are ..."<<endl;
 while(seqfile.read((char *)&Records,sizeof(Records)))
 {
   if(Records.rollno!=-1)
   {
```

```
   cout<<"\nName: "<<Records.name<<flush;
   cout<<"\nRoll No: "<<Records.rollno;
   cout<<"\nDivision :"<<Records.div;
   cout<<"\nAddress: "<<Records.address;
   cout<<"\n";
    }
 }
 seqfile.close();
}
int student::Search()
{
 fstream seqfile;
 int id,pos,offset;
 cout<<"\n Enter the roll no for searching the record ";
 cin>>id;
 seqfile.open("stud.DAT",ios::in|ios::binary);
 pos=-1;
 seqfile.seekg(0,ios::beg);
 int i=0;
 while(seqfile.read((char *)&Records,sizeof(Records)))
 {
  if(id==Records.rollno)
  {
   pos=i;
   break;
  }
  i++;
 }
  seqfile.close();
  return pos;
}
void student::deletion()
```

```
{
    int id,pos;
    cout<<"For deletion"<<endl;
    fstream seqfile;
    pos=Search();
    seqfile.open("stud.DAT",ios::in|ios::binary|ios::out);
    seqfile.seekg(0,ios::beg);
    if(pos==-1)
    {
    cout<<"\n Record is not present in the file";
    return;
    }
  int offset=pos*sizeof(Records);
  seqfile.seekp(offset);
  strcpy(Records.name,"");
  Records.rollno=-1;
  Records.div=-1;
  strcpy(Records.address,"");
  seqfile.write((char *)&Records,sizeof(Records));
  seqfile.seekg(0);
  seqfile.close();

}
int main()
{
student e;
 char ans='y';
 int choice,key;
 int h=0;
 do
 {
     cout<<"1.Create"<<endl;
```

```
    cout<<"2.Display"<<endl;
    cout<<"3.Search"<<endl;
    cout<<"4.Delete"<<endl;
    cout<<"Enter your choice"<<endl;
    cin>>choice;
        switch(choice)
        {
        case 1:
           e.Create();
           break;
        case 2:
          e.Display();
          break;
        case 3:
           h=e.Search();
           if(h<0)
           cout<<"\n Student not present in file"<<endl;
           else
           cout<<"\n Student is present in file"<<endl;
            break;
        case 4:
           e.deletion();
           break;
        }
        cout<<"Do you want to continue"<<endl;
        cin>>ans;
 }while (ans=='y');
return 0;
}
```

OUTPUT:-

CONCLUSION:-

We have successfully implemented file handling and performed functions like insertion,deletion and display of record using sequential file.