Unit-I
Object Oriented Programming in C++

Data File Handling In C++ Chapter: 05

File: - The information / data stored under a specific name on a storage device, is called a file.
Stream: - It refers to a sequence of bytes.
Text file: - It is a file that stores information in ASCII characters. In text files, each line of text is terminated with a special character known as EOL (End of Line) character or delimiter character. When this EOL character is read or written, certain internal translations take place.
Binary file:- It is a file that contains information in the same format as it is held in memory. In binary files, no delimiters are used for a line and no translations occur here.

Classes used for different file related operation

ofstream: Object of ofstream class used to write data to the files.
ifstream: Object of ifstream class used to read data from files
fstream: Object of fstream class used to both read and write from/to files.

Opening a file

Opening file using constructor

```
ofstream outFile("sample.txt");        //output only
ifstream inFile("sample.txt");         //input only
```

Opening File Using open ()

```
StreamObject.open("filename", [mode]);
ofstream outFile;
outFile.open("sample.txt");
ifstream inFile;
inFile.open("sample.txt");
```

File mode parameter	Meaning
ios::app	Adds data to the end of file
ios::ate	Cursor goes to end of file on opening and can be placed anywhere in the file.
ios::binary	File opens in binary mode
ios::in	Opens file for reading only
ios::out	Opens file for writing only and delete previous data if file already exists.
ios::nocreate	Open fails if the file does not exist
ios::noreplace	Open fails if the file already exist
ios::trunc	Deletes the contents of the file if it exist

All these flags can be combined using the bitwise operator OR (|). For example, if we want to open the file example.dat in binary mode to add data we could do it by the following call to member function open():

```
fstream file;
file.open ("example.dat", ios::out | ios::in | ios::binary);
```

Closing File

```
outFile.close();
inFile.close();
```

Input and output operation

put() and get() function

the function put() writes a single character to the associated stream. Similarly, the function get() reads a single character from the associated stream.

example :

file.get(ch);

file.put(ch);

write() and read() function

write() and read() functions write and read blocks of binary data.

example:

file.read((char *)&obj, sizeof(obj));

file.write((char *)&obj, sizeof(obj));

Determining End of File.

eof():-returns true (non zero) if end of file is encountered while reading; otherwise return false(zero)

File Pointers And Their Manipulation

All I/O stream objects have, at least, one internal stream pointer: ifstream has a pointer known as the get pointer that points to the element to be read in the next input operation.ofstream has a pointer known as the put pointer that points to the location where the next element has to be written.*fstream, inherits both, the get and the put pointers.* These internal stream pointers that point to the reading or writing locations within a stream can be manipulated using the following member functions:

seekg()	moves get pointer(input) to a specified location
seekp()	moves put pointer (output) to a specified location
tellg()	gives the current position of the get pointer
tellp()	gives the current position of the put pointer

The other prototype for these functions is:

seekg(offset, refposition);

seekp(offset, refposition);

The parameter offset represents the number of bytes(any negative or positive integer value for backward or forward movement) the file pointer is to be moved from the location specified by the parameter refposition. The refposition takes one of the following three constants defined in the ios class.

ios::beg start of the file

ios::cur current position of the pointer

ios::end end of the file

Program to count number of words from a text file"input.txt"

```
#include<fstream.h>
void main()
{       ifstream fin;
        fin.open("input.txt");
        char words[50]; int count=0;
        while(!fin.eof())
```

Program to count number of vowels in a text file "input.txt"

```
#include<fstream.h>
void main()
{
        ifstream fin;
        fin.open("input.txt");
        char ch; int count=0;
```

```
        { fin>>words;
        count++;
        }
        cout<<"Number of words in file is "<<count;
        fin.close();
}
```

```
        while(!fin.eof())
        {
                fin.get(ch);
 if(ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u')
                count++;
        }
cout<<"Number of vowels in file are "<<count;
fin.close();}
```

Binary File input output and modify record

```
#include<fstream.h>
#include<string.h>
#include<stdio.h>
class employee
{
int empno;
char ename[20];
public:
void getdata()
{
cout<<"\nEnter The Employee no. ";
cin>>empno;
cout<<"\n\nEnter The Name of The Employee ";
gets(ename);
}
void showdata()
{
cout<<"\nEmployeee no. : "<<empno;
cout<<"\nEmployee Name : ";
cout<<ename;
}
int returnempno()
 {
 return empno;
}
};
```

```
void write_data()
{
        employee obj;
        ofstream fout;
        fout.open("employee.dat",ios::binary|ios::app);
        obj.getdata();
        fout.write((char*)&obj,sizeof(obj));
        fout.close();
}
void display()
{
employee obj;
ifstream fin;
fin.open("employee.dat",ios::binary);
while(fin.read((char*)&obj,sizeof(obj)))
 {
 obj.showdata();
 }
         fin.close();
}
void modifyrecord(int n)
{
fstream finout;
student obj;
int found=0;
finout.open("employee.dat",ios::in|ios::out|ios::binary);
while(finout.read((char*)&obj,sizeof(obj))&&
found==0)
  {
        if(obj.returnempno()==n)
                {
                 obj.showdata();
                 cout<<"\nEnter The New data of employee";
                 obj.getdata();
                 int pos=-1*sizeof(obj);
                 finout.seekp(pos,ios::cur);
                 finout.write((char*)&obj,sizeof(obj));
```

```
                                found=1;
                                }
                        }
finout.close();
}
```

Short Answer Type Questions(2 Marks based on Text Files)

1.Write a function in a C++ to count the number of lowercase alphabets present in a text file "BOOK.txt".

Ans :-
```
int countalpha()
{
        ifstream Fin("BOOK.txt");
        char ch;
        int count=0;
        while(!Fin.eof())
        {       Fin.get(ch);
        if (islower(ch))
        count++;
        }
        Fin.close();
        return count;
}
```

2. Write a function in C++ to count the number of line started with alphabet 'a' or 'A' in a text file "LINES.TXT".

Ans. :-
```
void counter( )
{
char Aline[80];
int Count=0;
ifstream Fin ("LINES.TXT");
while(!fin.eof())
{
Fin.getline(Aline,80, '\n'));
if (Aline[0]== 'A'||Aline[0]=='a')
Count++;
}
cout<<Count<<endl;
Fin.close( );
}
```

3. Write a function to count number of words whose word length is 8, in a file named "article.txt".

Ans:
```
void wordlen8( )
{
char word[20];
  int count=0;
  ifstream fin("article.txt");
  while(!fin.eof())
        {
        fin>>word;
```

```
        if(strlen(word)==8)
                count++;
        }
    cout<<"Total number of words with word length 8 is=" << count; fin.close();
    }
```

4. Given a binary file PHONE.DAT, containing records of the following structure type.

```
class phonlist
{ char Name[20] ;
char Address[30] ;
char AreaCode[5] ;
char PhoneNo[15] ;
public :
void Register( ) ;
void Show( ) ;
int CheckCode(char AC[ ])
{ return strcmp(AreaCode, AC) ;
}
} ;
```

Write a function TRANSFER( ) in C++, that would copy all those records which are having AreaCode as "DEL" from PHONE.DAT to PHONBACK.DAT.

Ans:

```
void transfer( )
{
ifstream Fin;
ofstream Fout;
Phonlist ph;
Fin.open("PHONE.DAT", ios::in | ios::binary);
Fout.open("PHONBACK.DAT", ios::out | ios:: binary);
while(Fin.read((char*)&ph, sizeof(ph)))
{
if(ph.check("DEL") == 0)
        Fout.write((char*)&ph, sizeof(ph));
}
Fin.close();
Fout.close();
}
```

5. Given a binary file STUDENT.DAT, containing records of the following classStudent type

```
class Student
{
char S_Admno[l0]; //Admission number of student
char S_Name[30]; //Name of student
int Percentage; //Marks Percentage of student
public:
void EnterData()
{
gets(S_Admno);gets(S_Name);cin>>Percentage;
}
void DisplayData()
```

```
        {
        cout<<setw(12)<<S_Admno;
        cout<<setw(32)<<S_Name;
        cout<<setw(3)<<Percentage<<endl;
        }
        int ReturnPercentage(){return Percentage;}
};
```

Write a function in C++, that would read contents of file STUDENT.DAT and display the details of those Students whose Percentage is above 75

Answer :-

```
void Distinction()
{
Student S;
fstream Fin;
Fin.open("STUDENT.DAT", ios::binarylios::in);
while(Fin.read((char*)&S, sizeof(Student))
if (S.ReturnPercentage()>75)
S.DisplayData( );
Fin.close();
}
```

6. Given a binary file STUINFO.DAT, containing records of the following structure type.

```
class STUDENT
  {
 int rollno;
 char Name[20] ;
char Address[30] ;
char PhoneNo[15] ;
public :
void enter( )
{
        cin>>rollno;
cin.getline(name,20);
cin.getline(address,30);
cin.getline(phoneno,15);
     }
   void display( )
   {
 cout<<"information of student is";
cout<<rollno<<name<<address<<phoneno;
   }
} ;
```

Write a function stu_write( ) in C++, that would write information of students in STUINFO.DAT

Very Short Questions (1 Mark based on file pointer)

1. **Observe the program segment given below carefully and fill the blanks marked as Line 1 and Line 2 using fstream functions for performing the required task. 1**

```
#include <fstream.h>
class Library
{
long Ano; //Ano – Accession Number of the Book
char Title[20]; //Title – Title of the Book
int Qty; //Qty – Number of Books in Library
public:
void Enter(int); //Function to enter the content
void Display(); //Function of display the content
void Buy(int Tqty)
{
Qty+=Tqty;
} //Function to increment in Qty
long GetAno() {return Ano;}
};
void BuyBook (long BANo, int BQty)
//BANo is  Accessionno of the book purchased
//BQty is Number of books purchased
{
fstream File;
File. open ("STOCK.DAT", ios: : binarylios: : inlios: : out);
int Position=–1;
Library L;
while (Position = = –1 && File. read ((char*) &L, sizeof (L)))
if (L. GetAno() = =BANo)
{
L. Buy (BQty); //To update the number of Books
Positions=File. tellg()–sizeof (L);
//Line 1: To place the file pointer to the required position.
————————————;
//Line 2: To write the object L on to the binary file
————————————;
}
if (Position==–1)
cout<<"No updation done as required Ano not found...";
File. Close();
}
```

Ans. : File. seekp (position, ios :: beg); // Line–1
File. write ((char *) & L, sizeof (L)); // Line–2