QUESTION SOLVING TIPS IN CBSE BOARD EXAM

TOPIC : FILE HANDLING

Question no. 4 in CBSE BOARD EXAM PAPER

Contains 3 parts/type questions – Total weightage in Exam (8 marks)

1. Based on seekg(), seekp(), tellg(), tellp() functions - 1 mark
2. Based on get(), getline() and >>(input operator) -3-4 mark
3. Based on read() and write() function -3-4 mark

Type 1 questions

Based on get (), getline() or input operator(>>)

The questions will be like to count or to display words, characters, lines from an already existing file.

- for characters use get() function
- for words use (>>) input operator
- for line use getline() function

QUESTION EXAMPLES:

Q1. Write a function in C++ to count the number of uppercase alphabets present in a text file " ARTICLE.TXT".

Q2. Write a function to count and print the number of complete words as "to" and "are" stored in a text file "ESSAY.TXT".

Q3. Write a function in C++ to display lines starting with alphabet 'A' or alphabet 'E' present in a text file "LINES.TXT".

STEPS TO BE FOLLOWED

- **START FUNCTION**
  Void countupper()

{

- **DECLARE STREAM AND OPEN FILE USING STREAM**
  ifstream fin(“ARTICLE.TXT”);

- **ACCORDING TO THE FUNCTION DECLARE VARIABLES AS NEEDED FOR EX. TO STORE COUNTING, TO READ WORD, LINE OR CHARACTER FROM FILE.**
  Char ch;
  Int ctr=0;

- **START WHILE LOOP**
  While(fin)
  {

- **READ DATA FROM FILE AS REQUIRED**
  fin.get(ch);

- **PROCESS AS REQUIRED**
  If(isupper(ch))
  {
  Ctr++;
  }
  }
  Cout<<”no. of upper case characters in file”<<ctr;

- **CLOSE THE FILE AND THEN FUNCTION**

  fin.close();
  }

Q1. Write a function in C++ to count the number of uppercase alphabets present in a text file “ ARTICLE.TXT”.

```
Void countupper()
{
ifstream fin(“ARTICLE.TXT”);
Char ch;
Int ctr=0;
while(fin)
{
```

```
fin.get(ch);
If(isupper(ch))
{
Ctr++;
}
}
Cout<<"no. of upper case characters in file"<<ctr;
fin.close();
}
```

Q2. Write a function to count and print the number of complete words as "to" and "are" stored in a text file "ESSAY.TXT".

```
void CWORDS( )
{
ifstream fin("ESSAY.TXT");
char st[20];
int count=0;
while(fin)
{
fin>>st;
if(strcmpi(st,"to") = =0 || strcmpi(st,"are")= =0)
count++;
}
cout<<"\nTotal 'to' & 'are' words = "<<count;
fin.close( );
}
```

Q3. Write a function in C++ to display lines starting with alphabet 'A' or alphabet 'E' present in a text file "LINES.TXT".

```
void DISPLINES( )
{
        ifstream fin("LINES.TXT");
        char str[100];

        while(fin)
        {
                fin.getline(st,100);
                if(str[0]=='A'||str[0]=='E'|| str[0]=='a'||str[0]=='e')
                {
                        Cout<<"\n"<<str;
```

```
            }
        }

        fin.close( );
}
```

TYPE 2 QUESTION

Will be based on files storing data in form of objects of class or structure variable formations. read() and write() functions will be used to read or write data of this type.

Prototype of read and write functions

```
streamname.read((char *)&objname, sizeof(objname));

streamname.write((char *)&objname, sizeof(objname));
```

The steps to solve this type of questions will be same as discussed in type 2 questions but the difference will be just to read data we will use read() function and to write data we will use write() function and to store that data we will use object of the class type for which we want to read or write data and to process data class's member function will be used.

Q1. Given a binary file "BUS.DAT", containing records of the following class bus type.

```
class bus
{
int bus_no;
char desc[40];
int distance; //in km
public:
void read( )
{       cin>>bus_no;
        gets(desc) ;
        cin>>distance;
}
void display( )
{       cout<<bus_no;
        puts(desc);
        cout<<distance;
}
int retdist( )
```

```
{       return distance;
}
};
```

Write a function in C++ that would read the contents of file “BUS.DAT” and display the details of those buses which travels the distance more than 100 km.

Solution:-

```
void disprecord()
{
        ifstream fin(“BUS.DAT”);
        bus obj;
        while(fin)
        {
                fin.read((char *)&obj,sizeof(obj));

                if(obj.retdist()>100)
                {
                        obj.display();
                }
        }
        fin.close();
}
```

Q2. Given a binary file Sports.dat, containing records of the following structure type:

```
Struct Sports
{
Char Event[20];
Char Participant[10][30];
};
```

Write a function in C++ that would read contents from the file Sports .dat and creates a file named Athletic.dat copying only those records from Sports.dat where the event name is “Athletics”.

Solution :-

```
void copyrecord()
{
Ifstream fin(“Sports.dat”);
ofstream fout(“Athletics.dat”)
Sports game;
while(fin)
{
        fin.read((char *)&game,sizeof(game));
        if(strcmpi(game.Event,”Athletics”)==0)
        {
                fout.read((char *)&game,sizeof(game));

        }
}
```

```
fin.close();
fout.close();
}
```

TYPE 3

Fill in the blanks using
Seekg(), seekp(), tellg(), tellp(), read(), write()

fstream	
ifstream	ofstream
seekg() tellg()	seekp() tellp()

tellg(),tellp() – used to get the current postion of file cursor/pointer

seekg(), seekp() - used to move/place the file cursor/pointer to a desired location

example:-

```
Int pos=fin.tellp();
```

Prototype of seekp(),seekg()

seekg(pos, mode)

Modes:

ios::beg to move file pointer from beginning(default file mode)
ios::end to move file pointer from end of file
ios::cur to move file pointer from current position of cursor

examples:

1. **To place the file pointer in beginning of file**

fin.seekp(0); or fin.seekp(0,ios::beg); (as ios::beg is default is it optional)

2. **To place file pointer 5 points back from the end of file**

fin.seekg(-5,ios::end);

3. **To place the file pointer 10 points forward from its current position**

```
fin.seekp(10,ios::cur);
```

4. **To place the file pointer at end of the file**

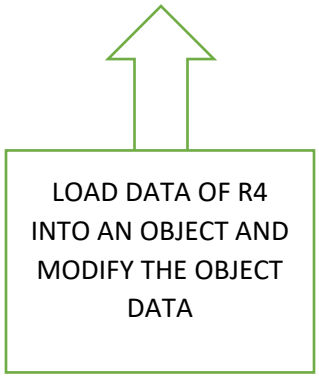
```
fin.seekg(0,ios::end);
```

THIS QUESTION WILL BE 99% ON MODIFY RECORD OF A FILE

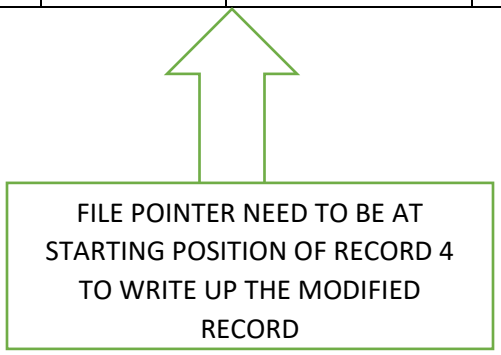
LET US FIRSTLY DICUSS THE PROCESS OF MODIFYING CONTENT OF A FILE

CONSIDER A FILE CONTATING RECORD OF 4 BITS EACH

RECORDNO	R1	R2	R3	R4	R5
SIZE	0-4	5-8	9-12	13-16	17-20



RECORDNO	R1	R2	R3	R4	R5
SIZE	0-4	5-8	9-12	13-16	17-20



AFTER PLACING THE FILE POINTER AT APPROPRIATE POSITION WE WILL CALL THE WRITE FUNCTION TO WRITUP THE MODIFIED RECORD BACK IN FILE

STEPS TO PERFORM THIS TASK IN CODING WAY

METHOD 1

```
Pos=fin.tellp();
Fin.read((char *)&obj,sizeof(obj));
Obj.modify();
fin.seekp(pos);
fin.write((char *),&obj,sizeof(obj));
```

METHOD 2

```
Fin.read((char *)&obj,sizeof(obj));
Obj.modify();
fin.seekp(-1*sizeof(obj),ios::cur);
fin.write((char *),&obj,sizeof(obj));
```

METHOD 3

```
Fin.read((char *)&obj,sizeof(obj));
Obj.modify();
fin.seekp(fin.tellg()-sizeof(obj));
fin.write((char *),&obj,sizeof(obj));
```

In fill in the blank question just check the name of object,stream name,stream type,and weather the programmer is storing the position of record prior to read a new record or not and then fill the blanks accordingly to method 1,2 or 3 as required.