DATA FILE HANDLING

CLASS XII

fstream.h

STREAM : A sequence of bytes.

Data File

A file to store data for later use.

It is of Two Types

TEXT FILE	BINARY FILE
Information stores in ASCII characters	In same format as it stored in memory
Each Line terminated (delimited) with a special character known as EOL (End of Line) character.	No termination for a line
Internal Translation required while read & write of EOL character.	No internal Translation
Due to translation tough to read & write	Faster and Easier to read & write than Text file
Requires more space in memory	Requires less space in memory than Text File

Opening of a File

There are Two different ways to open file

1. Using Constructor Function
2. Using open()

There are three different constructors of stream classes

(i)	ifstream	(to open input file)
(ii)	ofstream	(to open output file)
(iii)	fstream	(to open input/output file)

Example of Opening file using Constructor

ifstream fin (“Datafile.dat”);

- Reading only

ofstream fout (“Datafile.dat”);

- Writing only

fstream finout (“Datafile.dat”);

- Reading & Writing both

Example of Opening file using OPEN()

```
ifstream    fin ;
fin.open(“Datafile.dat”);                reading

ofstream    fout ;
fout.open(“Datafile.dat”);               writing

fstream    finout ;
finout.open(“Datafile.dat”);             both
```

File Mode

ios::in - Opening file for reading.

ios::out - Opening file for writing.

ios::app - Opening file to append record at end.

ios::ate - Opening file & seeking EOF.

ios::binary - Opening file as Bina

ios::nocreate - Opening file if exist.

ios::noreplace - Opening file if not exist (Create + Open).

Otherwise fail

Example of Opening file using OPEN() with open mode

```
ofstream    fout ;
fout.open(“Datafile.dat”, ios::binary | ios::out);
```

writing data to file

```
ofstream    fout ;
fout.open(“Datafile.dat”, ios::binary | ios::app);
```

writing data to file at end

Example of Opening file using OPEN() with open mode

```
ifstream    fin ;
fin.open(“Datafile.dat”, ios::binary | ios::in);
```

reading data from file

```
fstream    finout ;
finout.open(“Datafile.dat”, ios::binary | ios::in|ios::out);
```

both

Example Program

read() and write()

fin.read((char*) & struct/class ,   sizeof(struct/class)

fout.write(char*) & struct/class ,   sizeof(struct/class)

Example Program

File pointer and Random Access

- seekg() seekp() : *get pointer*
- tellg() tellp() : *put pointer*

- ios::beg : *beginning of file*
- ios::cur : *current position in the file*
- ios::end : *end f file*

example

- fin.seekg(0) ; : Start of file
- fin.seekg(0, ios::beg); : same as above
- fin.seekg(30, ios::beg); : 30 byte after from start

- fin.seekg(0, ios::end); : at end of file
- fin.seekg(-30, ios::end); : 30 byte previous

- fin.seekg(5, ios::cur); : 5 byte after current
- fin.seekg(-5, ios::cur); : 5 byte previous

Basic Operation on File

- Insert
- Display
- Search eg. with Key field With Record No.
- Modify example
- Delete example

Thanks………

Prepared by : DINESH PATEL
PGT [COMPUTER SCIENCE]
KENDRIYA VIDYALAYA,
KARANJA