Assignment 10

File Handling

Name: Aditya Kangune	DOP:
Roll NO: 23365	DOS:

Title: File Handling.

Aim: Implementation of sequential file.

Problem Statement: Department maintains Student's database. The file contains roll number, name, division and address. Write a program to create a sequential file to store and maintain student data. It should allow user to:

a.) Create a student database.

b.) Add a information of student.

c.) Delete information of student.

d.) Search and display information of particular student.

i.) If record is found, then display it.

ii.) If not found, then dispaly a message.

e.) Display records of all students in tabular form.

Theory:

1.) What is file data structure?

→ File Structure is the organization of data in secondary storage device in such a way that it minimizes the access time and storage space.

2.) Need for file data structure.

1.) Storing a file will preserve your data even if the program terminates.

2.) We can easily access contents of file using some commands.

3.) We can easily move data from one file to another.

3.) Types of file:

a.) Data and code file:

→ A data file is a computer file which stores data to be used by a computer application or system, including input and output data.

→ It usually does not contain instructions or code to be executed.

b.) Variable vs Fixed length files:

→ Fixed length records: All records in file have same size.

→ Variable length record: Different records in files have different sizes.

c.) Text vs Binary files:

→ Text files contain textual information in the form of alphabets, digits and special characters or symbols

→ Binary files contain bytes or a compiled version of a text file.

d.) Based of data organization:

→ Sequential files: contains and stores data in chronological order.

→ Index Sequencial files: Records are stored in the order that they are written to the disk.

→ Direct access files: All records are stored in direct access storage device (DASD), such as hard disk, randomly throughout the file.

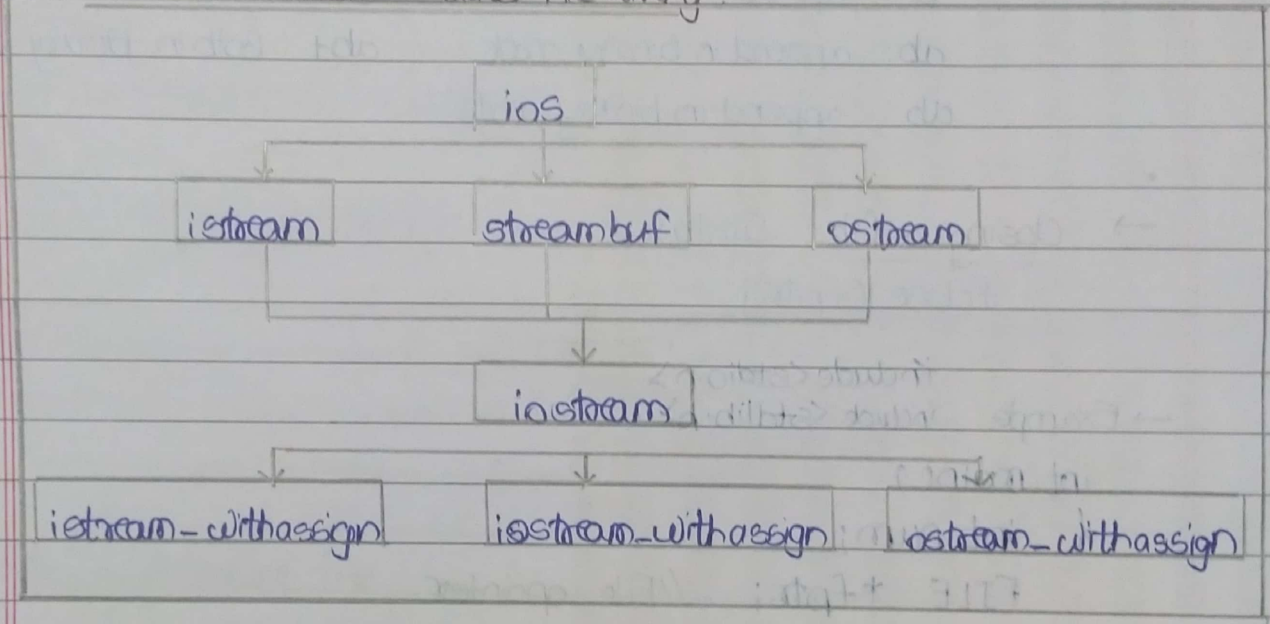
4.) File application:

Read, store, update data on a file.

5.) List daon various operations on the file:

- → Create → Appad
- → Delete → Seek
- → Open → Get attribute.
- → Close → Set attribute.
- → Read → Rename.
- → Write etc.

6.) C++ File basic : class hierarchy:



1.) ios class is the topmost class in the stream classes hierarchy. It's the base for istream, ostream and streambuf class.

2.) istream and ostream serves the base classes for iostream class. The class istream is used to for input and ostream for output.

3.) Class ios is indirectly inherited to iostream class using istream and ostream.

4.) The _withassign classes are provided with extra functionality for the assignment operations.

7.) File open with modes, close, read, write syntaxes in C++ with example:

→ Opening a file: Syntax: prt

ptr = fopen ("fileopen" , "mode");

path

→ □ Modes for file opening:

r	: Reading	r+	: Both read and write
rb	: Reading in binary mode	rb+	: Both in binary mode.
w	: writing	w+	: Both read and write.
wb	: writing in binary mode	wb+	: Both in binary mode.
a	: append	a+	: Both read and write.
ab	: append in binary mode	ab+	: Both in binary mode.
ab	: append in binary mode		

→ Closing a file: Syntax:

fclose (fptr);

```
         include <stdio.h>
→ Example: include <stdlib.h>
  int main()
  { int num;
    FILE *fptr;   //File epointer.
    // Opening to write
    fptr = fopen("c:\\example.txt", "w+");
    fprintf (fptr, "%d", 2020); / Adding text
    // Reading
    fscanf (fptr, "%d", &num);
    printf (num); // printing text in file
    // Closing the file.
    fclose(fptr);
    return 0;
  }
```

8.) isopen, eof meaning:

→ ifstream::is-open ⇒ returns if the stream is currenty associated to a file.

→ ios::eof ⇒ returns true if the eof bit error state flag is set for the stream.

9.) file pointer reposition functions - tellg, tellp, seek, etc:

→ tellg(): used to ~~move~~ know where the pointer is in the filp.

→ tellp(): get position in output sentence.

→ seekg(): used to move pointer to desired location w.r.t. reference pointer.

→ seekp(): used to move the put pointer to a desired location w.r.t a reference point.

Algorithms/Pseudocode:

a.) Create a student database: