**•Problem Statement :**

A shop maintains an inventory of items. It stores information of items like Item\_Code, Item\_Name, Quantity and Cost in a data file. Whenever Customer wants to buy an item, sales person inputs the Item\_Code and/or Item\_Name and the system searches in a file and displays whether it is available or not otherwise an appropriate message is displayed. If it is, then the system displays the item details and request for the quantity of items required. If the requested quantity of items are available, the total cost of items is displayed; otherwise the message is displayed as required items not in stock. After purchasing an item, system updates the file.

Design a system using a class called Items with suitable data members and member functions. Implement C++ program for the inventory system that will create a data file containing the Record of Items in the following form:

!~tab

tab~!

|  |  |  |  |
| --- | --- | --- | --- |
| **Item\_Code** | **Item\_name** | **Quantity** | **Cost in Rs.** |
| 3 | Pens | 24 | 10 |
| 17 | Notebooks | 46 | 14.99 |

**!~tab**

**•Data Members :**

!~s1Item\_Code

!~s1Item\_Name

!~s1Quantity

!~s1Cost

**•Member Function :**

!~s2Create file and store Record of Items

!~s2Search an Item in the file by Item\_Code or Item\_Name

!~s2Arrange the Items by Item\_Code or Item\_Name

!~s2Update the file

tab~!

**•Objectives :**

!~s3To learn the concepts of file handling in C++.To know to read and write text in the file.

!~s3To learn the file streams, file pointers and input and output operations on file.

!~s3To learn update operations on file.

**•Theory:**

In C++, files are mainly dealt by using three classes fstream, ifstream, ofstream available in fstream headerfile.

!~tab

•ofstream: Stream class to write on files

•ifstream: Stream class to read from files

•fstream: Stream class to both read and write from/to files.

tab~!

All the above three classes are derived from fstreambase and from the corresponding iostream class and they are designed specifically to manage disk files.

C++ provides us with the following operations in File Handling:

!~tab

•Creating a file: open()

•Reading data: read()

•Writing new data: write()

•Closing a file: close()

tab~!

Every file maintains two pointers called get\_pointer (in input mode file) and put\_pointer (in output mode file) which tells the current position in the file where reading or writing will takes place.

These pointers help attain random access in file. That means moving directly to any location in the file instead of moving through it sequentially.

# •**Algorithm / Class Diagram / Implementation :**

!~s4START.

!~s4Create the object of ofstream class.

!~s4Open the data file containing item information using ofstream class object.

!~s4Write the item information (Item\_Code, Item\_Name, Quantity and Cost) in a data file using output functions.

!~s4Close the file.

!~s4Create the object of ifstream class.

!~s4Open the data file using that object and read the records in a buffer from the file till eof().

!~s4Ask the user to search a record from a data file by Item\_Code or Item\_Name.

!~s4Input Item\_Code or Item\_Name from user to search Record.

!~s4Compare User’s Item\_Code or Item\_Name with Records in a data file which is read in a buffer.

!~s4If Matching found then display the complete records of Item (Item\_Code, Item\_Name, Quantity and Cost).

!~s4Else Display Message that Record not found.

!~s4 Ask the user to input quantity of required items.

!~s4 Compare User’s quantity with available quantity.

!~s4If the requested quantity of items are available, the total cost of items is displayed and update the records of items after purchase.

!~s4Else display the message as required items not in stock.

!~s4Arrange the Records in a file by Item\_Code or Item\_Name

!~s4Close the file.

!~s4STOP

# •**Platform :**

!~tab

•64-bit Open source Linux or its dderivatives.

•Open Source C++ Programming tool like G++/Eclipse Editor.

tab~!

**•Input :**

!~s5Item information (Item\_Code, Item\_Name, Quantity and Cost) to store in a data file.

!~s5Item\_Code/ Item\_Name to search a record in the file.

!~s5Quantity of required items to be purchased.

**•Output :**

!~s6Item information (Item\_Code, Item\_Name, Quantity and Cost)

!~s6Cost of Item to be purchased.

!~s6Updated Records in the File.

!~s6Sorted Records in the File.

**•Conclusion :**

Thus, implemented the Inventory System using File Handling concepts. This System is able to search the record by an Item\_code and Item\_Name. Records in the system are updated after purchase. Also All Records are arranged by Item\_code or Item\_Name.

**•FAQs :**

!~s7Which are the classes are used in file handling?

.>In C++, files are mainly dealt by using three classes fstream, ifstream, ofstream available in fstream headerfile.

!~s7Explain the file pointers and seek() and tell() functions?

.>Every file maintains two pointers called get\_pointer (in input mode file) and put\_pointer (in output mode file) which tells the current position in the file where reading or writing will takes place.

seekg() is a function in the iostream library that allows us to seek an arbitrary position in a file. It is mainly used to set the position of the next character to be extracted from the input stream from a given file in C++ file handling.

The tellg() function is used with input streams, and returns the current “get” position of the pointer in the stream. It has no parameters and returns a value of the member type pos\_type, which is an integer data type representing the current position of the get stream pointer.

!~s7Which are the functions used in read operation?

.> open(), read() and close() functions are used in read operation.

!~s7Write the File Opening Modes.

.> There are the following File Opening modes :

ios\_base::app - Seek to end-of-file before each write.

ios\_base::ate - Seek to end-of-file immediately after opening the file, if it

exists.

ios\_base::binary - Open file in binary mode (alternative is text mode).

ios\_base::in - Open file for input (implied for istream).

ios\_base::out - Open file for output (implied for ostream).

ios\_base::trunc - Truncate file, if it exists (default for ostream).

!~s7What is Eof()?

.>C++ provides a special function, eof( ), that returns nonzero (meaning TRUE) when there are no more data to be read from an input file stream, and zero (meaning FALSE) otherwise.