**Billing Management Project**

**COMPONENTS USED:**

Following are the components used for the completion of the project i.e;

* Structure
* Array
* Methods/Functions
* Pointer
* File Handling/ File Management
* Template

And Following are the libraries included to achieve the final desired output

* #include<iostream>
* #include<cstdlib>
* #include<string.h>
* #include<fstream>

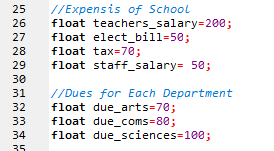
**DESIGN & ANALYSIS ALGORITHM:**

We’ve took 3 departments as a sample so that every record is saved and manipulated data.

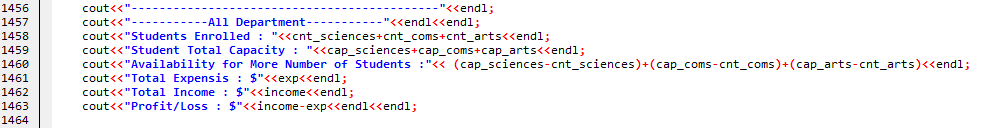
The sorting algorithm used in this program is **Insertion Sort**. It contains a nested loop for the sorting of the record with respect to the name of the students in record.

**STATISTICS OF THE SCHOOL:**

The statistics of the school is managed by the difference of incoming dues/fees of the students and the expenses of the school which includes **Salary of Teachers, Electricity Bill, Staff’s Salary and Tax.**



These are the variables where the default values are saved which are also changeable as mentioned in the requirements of question. The stats are calculated in the function named **void current\_details()**. This function shows the number of students and the remaining availability of students and also the profit loss ratio by calculating the difference of the incoming and the expenses of the school as mentioned above.



Change in the value of the any of the parameter will cause the change in the stats of the school and obviously effect the profit/loss ratio calculated in the end.

**WORKING & STRUCTURE OF PROGRAM:**

This program consist of 2 parts in the beginning,

* Student Record
* School Management

**STUDENT RECORD:**

Student Record consist of all the features with which we can retrieve and manipulate the existing data and can also enter new data in the record. Following are the functions, Student Record contains.

* **Print All Entries**

This function will print all the record present in our file in groups with respect to Departments of students. Basically all the records are saved in the array of structure and all three departments have their own array. So this function will print the record of all array of structure one by one.

* **Insert a New Record**

To insert new record, the program will ask for the department of the newly enrolled student. This thing will help in manually entering of record, selecting a department will automatically saves the name of department and will automatically choose that in which array it has to be saved. And admin does not have to enter the monthly fee of that student, program will detect the monthly due by itself from the saved dues of departments, rather admin just have to add if the students dues are paid or not,

* **Delete a Record**

As already discussed that the data is first saved in the Array of Structure and then it’s saved in file. So it is not possible to delete a data from the array. So to manage this issue, the data to be deleted is overwritten by the data present after it. In other words, all the data are shifted one step leftwards starting from the index number of the data which is to be deleted till the last index number. And the count of the student is also reduced by one so that the new data would overwrite the last data present which wasn’t deleted but was shifted one step leftwards.

* **Search for Record**

As we know that we need a unique key to search a record. So we can take a Roll Number/ registration Number as a unique key. But one thing coming in way was that, every department has that Roll Number. Let’s say if we want a Roll Number 1, then we will get 3 records of Roll Number 1, one for each department. So for that purpose, the program asks for the department of student. So that the program will search for that specific Roll Number in that specific department.

**A template** is used to write a function/class independent of a

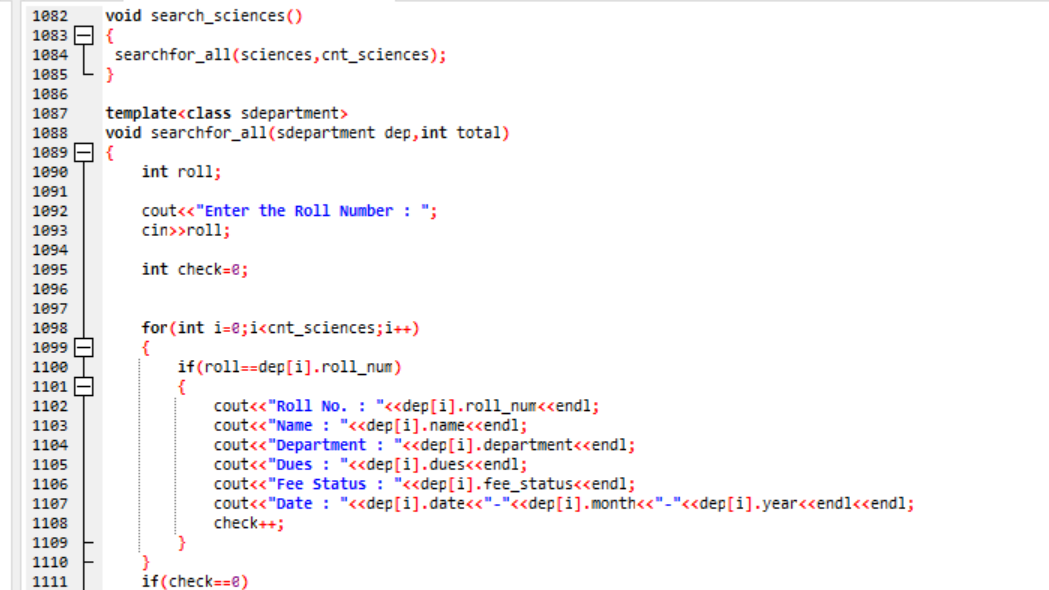
datatype. Since there is a similar way to search for student

information, we can use a template to do all the searching in

similar manner.

A generic function is used and called the other functions to the

generic function.



* **Update Existing Student Record**

Updating a record is a combination of Searching a Record and Entering a New Record. In the beginning, the program will ask for the Name of Department and the Roll Number of the students whose data is to be updated. And then shows the record of that student so that you can see what the previous saved record of that student is. After that, program will require the updated data of the student and will overwrite all the data on previous one.

* **Sort Record**

Sorting of the records is not in the groups of department as done earlier. The data of all the array of structure is first saved into the single array named **all[100]** by passing to the function named as **void merge().** Now this array is passed to the function named **sort()** using pointer. The array is passed as a pointer to the **sort()** function with the size of total records in integer.

**Sort()** then will sort the whole record **with respect to name of students**. And the algorithm used for this purpose is **insertion sort.**

**SCHOOL MANAGEMENT:**

School Management is programmed to check the current stats of the school and change the existing expenses of the school as well as the dues and capacity of number of students. This part leads to 2 branches.

* **Check Current Details**

This option will lead to the page where all the details are present in the group of each department individually and also collective on the school level. This Contains following details

* + - Students Enrolled in Each Department
    - Total Capacity of Students in Each Department
    - Availability of more Students in Each Department
    - Fee of each Department
    - Total Expenses of the School
    - Total Income of School
    - Profit/Loss Ratio
* **Change Existing Details**

This options let the admin to manipulate the existing data of the school management. For Example, Admin can change the Fee Structure of any department of school. Which will affect the whole statistics of school in the form of total income and profit/loss ratio. The changes admin can make are as follows.

* + - Change Dues of Departments
    - Change Salary of Teachers
    - Change Salary of Staffs
    - Change Capacity of Students

Before changing any of the details, the program will show the previous saved data so that admin can change details accordingly without any mistake.

* **File management**

We used file management as a mechanism to store the output of a program in a file and perform various operations on it. So we stored our data permanently on secondary storage (disk) and so that it can be retrieved when needed.

