A Project Report

On

**School billing system in ‘C’ language**

A Report submitted in partial fulfillment of the requirement of MCA Winning Camp

**CHANDIGARH UNIVERSITY GHARUAN, PUNJAB**

By

**Garvesh Garg (21MCI1222)**

**ACKNOWLEDGEMENT**

This is to acknowledge that this project titled ***‘School Billing System in ‘C’ Language’*** is the result of co-operation and guidance of prominent persons in the subject line. I would Thank ***Dr. Manisha Malhotra (Head of Department UIC), Chandigarh University*** for his great support. I am sincerely thankful to honorable guides for providing me with the necessary facilities to carry out the project successfully. This is only because of their valuable guidance that I am able to submit this project with all its requirements duly fulfilled.

I would also thanks to all teaching and non-teaching staff of U.I.C., involved directly or indirectly in making the project a grand success.

And finally, I would like to thanks ***Chandigarh University*** for providing me such a great opportunity of exploring and shaping my presentation skills as a part of our course.

**NAME: GARVESH GARG**

**UID: 21MCI1222**

**COURSE: MCA (AI & ML)**

**Certificate**

This is to certify that this is a bona fide record of the project work done satisfactorily at ***University Institute of Computing*** by ***GARVESH GARG (UID: 21MCI1222)*** in partial fulfillment of ***M.C.A.*** (Winning Camp) & ***CHANDIGARH UNIVERSITY PUNJAB.***

This report or a similar report on the topic has not been submitted for any other examination and does not form part of any other course undergone by the candidate.

**(Sign of Coordinator) (Sign of Candidate)**

**(Ms. Rhythm Beri) Chandigarh University,**

**Place: Gharuan, Punjab Gharuan, Punjab**

**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**PROJECT PROPOSAL CONTENTS**

Table of Contents:-

1) Introduction of the project

2) Problem Analysis

3) Project Analysis

4) Cost & Benefit Analysis

5) Feasibility Study

6) Data Dictionary

7) Project Category

8) Tools/Platform, Hardware and Software requirement specification.

9) Coding

10) Output Screenshots

**Introduction**

A System can work smoothly if and only if all parts are work according to schedule or predefined standards gives, So we are developing the software for a ***“School Billing System in ‘C’ Language”***.

The term system is derived from the Greek word, which means as organized relationship among functioning units or components. Here we are presenting computer based information system. There are many types of systems. In our life many systems are working in a systematic way to performing some specific task, it is a system. In this computer base system the information system relies on the computer for handling business application.

The computer is now a required source of information. System relies heavily on computers for problem solving. This suggests that user must be familiar with computer technologies and have experience in handling people in an organization context.

So We are representing a computer base ***“School Billing System”*** to reduce all errors or problems that are arises in the manual billing system in schools. It works in an organizational manner, and properly managed fee/salary and all the information about the student/Teacher/Staff can easily gettable, like student/teacher/staff’s information, about fee/salary etc.

This project is complete and totally error-free. It utilizes file handling and data structures.

**Problem Analysis**

The **“*School Billing System”*** depends heavily on the institution’s needs and specifications on the types of information supplied by the guidance to the students. Thus another major area, which needs heavy consideration, is the information gathering phase of the product development.

**Information Gathering: Sources and Techniques**

In order to gain a sound understanding of the current process and the exact requirements of the employees, we undertook the method of “on-site-visits” and “interactions” with the administrator of the company’s managements. We observed the actual management process take place. The administrators were kind enough to answer my ‘ Whys ’ , ‘ How’s ’ and ‘ What’s ’ in this regard. In order to estimate the volume and type of data to be recorded, we studied the actual records of the institution. This gave us an in-depth idea and through insight into the design and structure of the database required to record the current process and the sub processes involved there in.

**Software Development Methodology Used:**

The product has been development using the Spiral model of the software development conspiring of:

1. Customer Communication 4. Engineering

2. Planning 5. Construction and Release

3. Risk Analysis 6. Customer Evolutions

**Project Analysis**

**System Analysis**

A system can be defined as “A set of arrangement of elements that are organized to accomplish some method, procedure or control by processing information.”

System Analysis stages down the broad frame work of strategy stage into it’s minute details. This is this is the stage where intensive user interaction assumes a lot of importance.

**System Development**

Development Phase, translates a set of requirements into an operational system that we call software. At early stages of hardware development a hardware engineer does not reach for a soldering iron, similarly the Software Engineer should not reach for the compiler. Design is accomplished first.

The first phase of development concentrates on design. Design phase for software with a description of architecture and data design. That is a modular structure is developed first, interfaces are defined and data structure is established. Design criteria are used for the assessment of quality.

This preliminary design step is reviewed for completeness and tractability to software requirements.

Procedural aspects of each modular component of the software design are considered next, and is following by the CODING – the generation of program using an appropriate programming language or CASE tool – occurs after the design is complete. Software engineering methodology views coding as a consequence of good design. Code is then reviewed for style and clarity, but should otherwise be directly traceable to a detailed design description.

**COST & BENEFIT ANALYSIS**

In developing cost estimates for a system, we need to consider servable cost elements among them hardware, personnel, facility, operating & supply costs.

**Hardware Cost**

Hardware cost relate to the actual purchase or lease of the computer & peripherals Ex. Printer, disk, CD drive etc. the best way to control for this cost is to treat it as an operating cost.

**Personnel Cost**

This includes staff salaries & benefits (health insurance, vocation time, sick pay etc.) as well as pay for those involved in developing system. Costs incurred during the development of a system are one times costs & are labeled development costs. Once the system is installed the costs of operating & maintaining the system become recurring costs.

**Facility Costs**

Facility costs are expenses incurred in the preparation of the physical site where the application or the computer will be I operation. This includes wiring, flooring, acoustics, lighting & air conditioning. These costs are treated as one times costs.

**Operation Costs**

Operation costs include all costs associated with the day to day operating system. The amount depends on the number of shifts the nature of the application & the calibers of the operating staff. There are various ways of covering operating costs to overhead. Another approach is to change each authorized user for the amount of processing they request from the system. The amount changed is based on the computer time, staff time & the volume of output produced.

**Supply Costs**

These costs are variable costs that increase with increased use of paper, ribbon, disks & the like they should be estimated & include in the overall cost of the system include rental & annual deprecation of any equipment such as typewriter, copies, computer terminals & microcomputers used in connection with costs associated with the equipment, such as insurance & maintenance should also be included.

**Equipment Cost**

Equipment cost include rental & annual deprecation of any equipment such as typewriter, copies, computer terminals & microcomputers used in connection with costs associated with the equipment, such as insurance & maintenance should also be included.

**FEASIBILITY STUDY**

Feasibility study before development or customization of software we think following of the major points, which included following:

**Economical**

This study about the software basically give the suggestion like if we develop a software for our organization than how much it will forfeitable, workable as well as economical. If we developing a software which very economical but not profitable there it is also wrong decision, So for we economical study about the system table following of the points:

How much costing of the software?

What is the output of the customization?

What are the advantages in compression manual?

We check:

Whether it is time consuming or not?

Whether it is use friendly or not?

It is mountable for long term.

Understand about the software.

Corrective of software.

**Technical**

We also consider some technical points about development of software.

We see:

Whether the software is reliable or not?

Whether it works for long time or not?

What are the languages as well as OS is used for development of software?

It provides security or not.

Software is reusable or not.

Software is updated or not.

Software is flexible or not. Whether software can maintain?

**Duration**

It is also a major point when we develop software. We also consider about time, how much time software place to complete because for a big organization time factor is money for full successful of the software also Tax sometimes so we have consider from starting to last that means beginning to software.

**DATA DICTIONARY**

Data dictionary is a structured repository of data about data. It is a list of terms and their definitions for all data items and data stores of a system.

The primary advantages that a data dictionary offers are as follows:

Documentation is one of the most obvious advantage of data dictionary.

It is a valuable reference in any organization.

It improves analysis/user communication by establishing consistence definitions of various elements, terms and procedures.

During implementation it serves as a common base against which programmers compare their data descriptions.

Programs that use a given data elements are cross referenced in the data dictionary, which makes it easy to identify them and make any necessary changes.

A Data Dictionary is an important step in building a database. Most data base management systems have a data dictionary as a standard feature.

**Project category**

This project is in ***‘C’*** language. In this project, we perform billing or accounting operations under two account types: one for the students and another for teachers and staffs.

This project is complete and totally error-free. It utilizes file handling and data structures.

School Billing System is a console application without graphic. The source code for this project is over ***1000*** lines.

**Features of project**

In this project, we can add, record, modify, search and delete the records of both account types. In addition to that, this mini project in ‘C’ language allows you to display fees, dues, total and advance of students, and salary-related information of teachers and staffs.

For the entry of records, current date and month is asked. Then, you can select the account type, and perform billing operations like I mentioned above. In add record, Name, Class and Roll No. of the student is asked, and it is similar for all other functions as well as the teacher’s account.

Data structures have been used effectively to handle co-related functions and store the record. This school billing system project comprises the following data structures:

**struct dat** – To store the date (month and day) of entry of records.

**struct student** – To store and organize the record of individual students.

**struct teacher** – To store and organize the record of individual teachers/staffs.

I have used the different functions for performing different billing operations in School Billing System. Following are some functions which will give you an outline of the project:

**start()** – Shows the account selection screen.

**chkdat()** – For checking date.

**addrec()** – For adding records.

**modrec()** – For modifying records.

**searchrec()** – For searching records.

**delrec()** – For deleting records.

**fee()** – For recording the fee paid and displaying fine, due, total and advance.

**salary()** – For calculating the salary of teachers and staffs.

**ext()** – For exiting.

This mini project on school billing system in C utilizes file handling in three of the functions mentioned above delrec(), fee() and salary().

**Tools/Platform, Hardware & Software Requirement Specification**

Software and Hardware specifications the name suggests, tells us about the various characteristics of the development environment. Here we are specifying various tools and languages that have been used in developing the application because of their relative ease of understand and also personal interest of the developing team.

**Processing Environment:** The project runs on every type of OS like Windows XP, 7, 8, 10, Mac OS etc.

**Configuration:** At least 60 GB Hard Disc, 512 MB RAM, Pentium ¾.

**Language:** ‘C’ Language.

**Editors:** Visual Studio/Dev C++/Turbo C++.

Coding

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<conio.h>

#include<windows.h>

struct dat//for date(month and day)

{

int d,m;//d=day,m=month

};

int clscanf();//check class (1-12)

struct student

{

struct dat dt;

float f,fine,tot,adv,due;//f=fee

char n[50];

int r,c;//roll and class

} stud,s;

struct teacher

{

struct dat dt;

char n[50];

float sal,adv,tot;

int id,no;

} tech,t;

int chkdat(int,int);// for checking date

void addrec(int);//for adding records

void modrec(int);//for modifying records

void searchrec(int);//for searching records

void delrec(int);//for deleting records

void salary(int);//for the calculation of salary of teacher and staff

FILE \*fs,\*ft;//file declaration

int mm,dd;//mm=month, dd=day

void ext();//for exiting

void main(void)

{

int i,j,k;

for(i=0; i<120; i++)

{

printf("\xdb");

}

printf("\n");

for(i=0; i<120; i++)

{

printf("\xdb");

}

system("color e1");

printf("\n\t\t\t\t\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*\n");

printf("\t\t\t\t| |\n");

printf("\t\t\t\t========================================================\n");

printf("\t\t\t\t|\*\*WELCOME TO C PROGRAM SCHOOL BILLING SYSTEM PROJECT\*\*|\n");

printf("\t\t\t\t========================================================\n");

printf("\t\t\t\t| DEVELOPED BY GARVESH GARG |\n");

printf("\t\t\t\t========================================================\n");

printf("\t\t\t\t| |\n");

printf("\t\t\t\t\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*=\*\n");

printf("\n");

for(i=0; i<120; i++)

{

printf("\xdb");

}

printf("\n");

for(i=0; i<120; i++)

{

printf("\xdb");

}

printf("\n\t\t\t\t\t PLEASE ENTER ANY KEY TO CONTINUE");

for(i=0; i<5; i++)

{

printf(".");

Sleep(500);//after printing one . another comes after 0.5 seconds

}

getch();

system("cls");//clears the screen

printf("\n");

for(i=0; i<120; i++)

{

printf("\xdb");

}

system("color 6d");

printf("\n\n\t\t\t\t\*\* WELCOME TO C PROGRAM SCHOOL BILLING SYSTEM PROJECT \*\*\n\n");

for(i=0; i<120; i++)

{

printf("\xdb");

}

printf("\n\n\t\t\t\t\t PLEASE ENTER ANY KEY TO START");

for(i=0; i<5; i++)

{

printf(".");

Sleep(500);

}

fflush(stdin);

getch();

system("cls");

system("color 0f");//1st is for back ground color and second is for text color

printf("\n PLEASE ENTER CURRENT DATE\n mm dd\n ");

scanf("%d%d",&mm,&dd);

mm=chkdat(mm,dd);

start();

}

void start()

{

int i,j;//j is for selection of account type

system("cls");

printf("\n\t\tPLEASE ENTER ACCOUNT TYPE");

printf("\n\t\t1:: Student");

printf("\n\t\t2:: Teachers and Staffs");

printf("\n\t\t3:: Exit");

printf("\n\t\tAccount type choice");

for(i=0; i<5; i++)

{

printf(".");

Sleep(500);

}

fflush(stdin);

scanf("%d",&j);

switch (j)

{

case 3:

ext();

case 1:

{

system("cls");

printf("\n\t\tPLEASE ENTER THE CHOICE");

printf("\n\t\t1:: Add record");

printf("\n\t\t2:: Search record");

printf("\n\t\t3:: Modify record");

printf("\n\t\t4:: Delete record");

printf("\n\t\t5:: Calculate fee");

printf("\n\t\t6:: Exit");

printf("\n\n Enter choice ");

fflush(stdin);

scanf("%d",&i);

switch (i)

{

case 1:

addrec(j);//function call

start();//function call

case 2:

searchrec(j);

start();

case 3:

modrec(j);

start();

case 4:

delrec(j);

start();

case 5:

fee(mm);

start();

case 6:

ext();

default :

{

printf("\n\n\tInvalid entry!!");

printf("\n\nTo Account Type\n\n\t");

system("pause");

start();

}

}

}

case 2:

{

system("cls");

printf("\n\t\tPLEASE ENTER THE CHOICE");

printf("\n\t\t1:: Add record");

printf("\n\t\t2:: Search record");

printf("\n\t\t3:: Modify record");

printf("\n\t\t4:: Delete record");

printf("\n\t\t5:: Calculate Salary");

printf("\n\t\t6:: Exit");

printf("\n\n Enter choice ");

fflush(stdin);

scanf("%d",&i);

switch (i)

{

case 1:

addrec(j);

start();

case 2:

searchrec(j);

start();

case 3:

modrec(j);

start();

case 4:

delrec(j);

case 5:

salary(mm);

start();

case 6:

ext();

default :

{

printf("\n\n\tInvalid entry!!");

printf("\n\nTo Account Type\n\n\t");

system("pause");

start();

}

}

}

default :

{

printf("\n\n\tInvalid entry!!");

printf("\n\nTo Account Type\n\n\t");

system("pause");

start();

}

}

}

void addrec(int j)

{

int dif,cdat,ddat,month=0;//cdat=month till which fee is cleared

float ff;//used in calculatin of fee of different class

char c='y';

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ADD RECORD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

if (j==1)

{

while(c=='y'||c=='Y')

{

int a=1;

printf("\n\nEnter the name of student: ");

fflush(stdin);

scanf("%[^\n]",stud.n);

printf("\nEnter the class: ");

fflush(stdin);

stud.c=clscanf();

printf("\nEnter the Roll No.:");

fflush(stdin);

scanf("%2d",&stud.r);

printf("\nEnter month and day till which fee is paid:");

fflush(stdin);

scanf("%2d%2d",&cdat,&ddat);

cdat=chkdat(cdat,ddat);

stud.dt.m=cdat;

ff=stud.c/10.0;

stud.f=1000\*(1+ff);//fee of different classes

dif=mm-stud.dt.m;//months of fee left to be paid

stud.fine=(dif\*stud.f)\*1/100;

stud.due=(dif)\*stud.f;//fees left to be paid

if(dif==1)

{

stud.tot=stud.f;

stud.fine=0;

}

else

{

stud.tot=stud.fine+stud.due;

}//for calculation of total fee

fs=fopen("student","ab+");//opening a binary file in apend mode

fwrite(&stud,sizeof(stud),1,fs);

fclose(fs);

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

if (j==2)

{

while(c=='y'||c=='Y')

{

int a=1;

printf("\n\nEnter name of teacher/staff:" );

fflush(stdin);

scanf("%[^\n]",tech.n);

printf("\nEnter teacher/staff id: ");

fflush(stdin);

scanf("%d",&tech.id);

printf("\nEnter number of class/shift per month:: ");

scanf("%d",&tech.no);

fflush(stdin);

printf("\nEnter month and day till which salary is paid::");

scanf("%d %d",&tech.dt.m,&tech.dt.d);

cdat=chkdat(cdat,ddat);

tech.dt.m=cdat;

tech.sal=tech.no\*500;

tech.adv=(tech.dt.m-mm-1)\*tech.sal;

if (tech.adv<0) tech.adv=0;

tech.tot=tech.sal;

ft=fopen("teacher","ab+");

fwrite(&tech,sizeof(tech),1,ft);

fclose(ft);

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

fflush(stdin);

printf("\n\n");

system("pause");

}

}

void searchrec(int j)

{

char name[50],namet[50];

int a=1,choice;

char c='y';

if (j==1)

{

while(c=='y'||c=='Y')

{

int a=1;

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SEARCH RECORD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\n\t\tPLEASE CHOOSE SEARCH TYPE::");

printf("\n\n\t\t1::Search by name::");

printf("\n\n\t\t2::Search by class::");

printf("\n\n\t\t3::Search by rollno::");

printf("\n\n\t\t4::Exit");

printf("\n\n\t\t::Enter your choice:: ");

fflush(stdin);

scanf("%d",&choice);

if (choice==1)

{

a=1;

printf("\n\nEnter name of student to search: ");

fflush(stdin);

scanf("%[^\n]",name);

fs=fopen("student","rb");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

if (strcmpi(name,stud.n)==0)

{

a=0;

printf("\nname = %s",stud.n);

printf("\nclass = %d",stud.c);

printf("\nroll no = %d",stud.r);

printf("\nmonthy fee =%.2f",stud.f);

printf("\nlast fee paid in month =%2d",stud.dt.m);

printf("\n due=%.2f",stud.due);

printf("\n fine=%.2f",stud.fine);

printf("\n total=%.2f\n\n",stud.tot);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(fs);

}

else if (choice==2)

{

int cl;

a=1;

printf("\n\nEnter class of student to search: ");

fflush(stdin);

cl=clscanf();

fs=fopen("student","rb");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

if (stud.c==cl)

{

a=0;

printf("\nname = %s",stud.n);

printf("\nclass = %d",stud.c);

printf("\nroll no = %d",stud.r);

printf("\nmonthy fee =%.2f",stud.f);

printf("\nlast fee paid in month =%2d",stud.dt.m);

printf("\n due=%.2f",stud.due);

printf("\n fine=%.2f",stud.fine);

printf("\n total=%.2f",stud.tot);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(fs);

}

else if (choice==3)

{

int rll;

a=1;

printf("\n\nEnter roll of student to search: ");

fflush(stdin);

rll=clscanf();

fs=fopen("student","rb");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

if (strcmpi(name,stud.n)==0)

{

a=0;

printf("\nname = %s",stud.n);

printf("\nclass = %d",stud.c);

printf("\nroll no = %d",stud.r);

printf("\nmonthy fee =%.2f",stud.f);

printf("\nlast fee paid in month =%2d",stud.dt.m);

printf("\n due=%.2f",stud.due);

printf("\n fine=%.2f",stud.fine);

printf("\n total=%.2f",stud.tot);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(fs);

}

else if(choice==4)

{

ext();

}

else

{

printf("\n\n\n\t\tINVALID ENTRY!!!!\n\n\t\t");

system("pause");

searchrec(1);

}

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

if (j==2)

{

while(c=='y'||c=='Y')

{

int a=1;

printf("\n\nname of teacher/staff to search: ");

fflush(stdin);

scanf("%[^\n]",namet);

ft=fopen("teacher","rb");

while(fread(&tech,sizeof(tech),1,ft)==1)

{

if (strcmp(namet,tech.n)==0)

{

a=0;

printf("\nname = %s",tech.n);

printf("\nteacher/staff id = %d",tech.id);

printf("\nmonth till when salary is paid =%d",tech.dt.m);

printf("\nmonthy salary = %.2f",tech.sal);

printf("\nadvance paid = %.2f",tech.adv);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(ft);

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

}

void modrec(int j)

{

char name[50];

int a=1,choice,cl,rolno;

char c='y';

if (j==1)

{

while(c=='y'||c=='Y')

{

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MODIFY RECORD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\n\t\tPLEASE CHOOSE MODIFY TYPE::");

printf("\n\n\t\t1::Modify by name::");

printf("\n\n\t\t2::Modify by name &class::");

printf("\n\n\t\t3::Modify by name,class & rollno::");

printf("\n\n\t\t4::Exit");

printf("\n\n\t\t::Enter your choice:: ");

fflush(stdin);

scanf("%d",&choice);

if (choice==1)

{

int a=0;

printf("\n\nenter name of student to modify: ");

fflush(stdin);

scanf("%[^\n]",name);

fs=fopen("student","rb+");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

a=1;

if (strcmpi(name,stud.n)==0)

{

a=0;

printf("\nenter new name of student: ");

fflush(stdin);

scanf("%[^\n]",stud.n);

printf("\nenter new class of student: ");

fflush(stdin);

stud.c=clscanf();

printf("\nenter new roll of student: ");

fflush(stdin);

scanf("%d",&stud.r);

fseek(fs,-sizeof(stud),SEEK\_CUR);

fwrite(&stud,sizeof(stud),1,fs);

fclose(fs);

}

}

if (a==1)

printf("\n\nRECORDS NOT FOUND");

else

printf("\n\nRECORDS SUCCESSFULLY MODIFIED");

printf("\n\n");

system("pause");

}

else if (choice==2)

{

int a=0;

printf("\n\nenter name of student to modify: ");

fflush(stdin);

scanf("%[^\n]",name);

printf("\n\nenter class of student to modify: ");

fflush(stdin);

cl=clscanf();

fs=fopen("student","rb+");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

a=1;

if (strcmpi(name,stud.n)==0 && cl==stud.c)

{

a=0;

printf("\nenter new name of student: ");

fflush(stdin);

scanf("%[^\n]",stud.n);

printf("\nenter new class of student: ");

fflush(stdin);

stud.c=clscanf();

printf("\nenter new roll of student: ");

fflush(stdin);

scanf("%d",&stud.r);

fseek(fs,-sizeof(stud),SEEK\_CUR);

fwrite(&stud,sizeof(stud),1,fs);

fclose(fs);

}

}

if (a==1)

printf("\n\nRECORDS NOT FOUND");

else

printf("\n\nRECORDS SUCCESSFULLY MODIFIED");

printf("\n\n");

system("pause");

}

else if (choice==3)

{

int a=0;

printf("\n\nenter name of student to modify: ");

fflush(stdin);

scanf("%[^\n]",name);

printf("\n\nenter class of student to modify: ");

fflush(stdin);

cl=clscanf();

printf("\n\nenter roll of student to modify: ");

fflush(stdin);

scanf("%d",&rolno);

fs=fopen("student","rb+");

while(fread(&stud,sizeof(stud),1,fs)==1)

{

a=1;

if (strcmpi(name,stud.n)==0 && cl==stud.c &&rolno==stud.r)

{

a=0;

printf("\nenter new name of student: ");

fflush(stdin);

scanf("%[^\n]",stud.n);

printf("\nenter new class of student: ");

fflush(stdin);

stud.c=clscanf();

printf("\nenter new roll of student: ");

fflush(stdin);

scanf("%d",&stud.r);

fseek(fs,-sizeof(stud),SEEK\_CUR);

fwrite(&stud,sizeof(stud),1,fs);

fclose(fs);

}

}

if (a==1)

printf("\n\nRECORDS NOT FOUND");

else

printf("\n\nRECORDS SUCCESSFULLY MODIFIED");

printf("\n\n");

system("pause");

}

else if (choice==4) ext();

else

{

printf("\n\n\n\t\tINVALID ENTRY!!!!\n\n\t\t");

system("pause");

modrec(1);

}

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

if (j==2)

{

while(c=='y'||c=='Y')

{

int a=1;

printf("enter name of teacher to modify: ");

fflush(stdin);

scanf("%[^\n]",name);

ft=fopen("teacher","rb+");

while(fread(&tech,sizeof(tech),1,ft)==1)

{

if (strcmpi(name,tech.n)==0)

{

a=0;

printf("\nenter new name of teacher: ");

fflush(stdin);

scanf("%[^\n]",tech.n);

printf("\nenter new id of teacher: ");

fflush(stdin);

scanf("%d",&tech.id);

fseek(ft,-sizeof(tech),SEEK\_CUR);

fwrite(&tech,sizeof(tech),1,ft);

fclose(ft);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

else

printf("\n\nRECORD SUCCESSFULLY MODIFIED");

printf("\n\n");

system("pause");

fflush(stdin);

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

}

void delrec(int j)

{

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DELETE RECORD \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

FILE \*temp,\*t1;

int a=1;

char name[50],c='y';

if (j==1)

{

while(c=='y'||c=='Y')

{

int a=1;

printf("\n\nenter name of student to delete: ");

fflush(stdin);

scanf("%[^\n]",name);

fs=fopen("student","rb");

temp=fopen("tempfile","wb");//opening of temporary file for deleting process

while (fread(&stud,sizeof(stud),1,fs)==1)

{

if (strcmp(stud.n,name)==0)

{

a=0;

continue;

}

else

{

fwrite(&stud,sizeof(stud),1,temp);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

else

printf("\n\nRECORD SUCCESSFULLY DELETED");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(fs);

fclose(temp);

system("del student");/\*all data except the data to be

deleted in student were 1st moved to temp and data in student

was deleted\*/

system("ren tempfile, student");//renaming temp to student

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

if (j==2)

{

a=1;

char namet[50];

while(c=='y'||c=='Y')

{

printf("\n\nEnter name of teacher to delete record: ");

fflush(stdin);

scanf("%[^\n]",namet);

ft=fopen("teacher","rb");

t1=fopen("tempfile1","wb");

while (fread(&tech,sizeof(tech),1,ft)==1)

{

if (strcmp(tech.n,namet)==0)

{

a=0;

continue;

}

else

{

fwrite(&tech,sizeof(tech),1,t1);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

else

printf("\n\nRECORD SUCCESSFULLY DELETED");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(ft);

fclose(t1);

system("del teacher");

system("ren tempfile1, teacher");

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

}

void salary(int mm)

{

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* SALARY \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

FILE \*f,\*t;

int a=1,day;

char name[50],c='y';

int month,dif,id;

while(c=='y'||c=='Y')

{

int a=1;

fflush(stdin);

printf("\n\nEnter name:: ");

scanf("%[^\n]",name);

printf("\n\nEnter ID:: ");

scanf("%d",&id);

f=fopen("teacher","rb+");

t=fopen("te","wb+");

while(fread(&tech,sizeof(tech),1,f)==1)//file opened

{

if(strcmp(tech.n,name)==0 )//name entered is compared to the existing name in file

{

float lsal;

a=0;

printf("\n\nEnter the month till which salary is to be paid:: ");

fflush(stdin);

scanf("%d",&month);

month=chkdat(month,day);

tech.adv=(month-mm-1)\*tech.sal;

if (tech.adv<0) tech.adv=0;

lsal=mm-tech.dt.m;//months of salary left to be paid

if(lsal<0) lsal=0;

tech.tot=tech.adv+tech.sal\*(1+lsal);

if(month==tech.dt.m) tech.tot=0;

printf("\nmonthy salary left to be paid:: %.2f",lsal);

printf("\ntotal :: %.2f",tech.tot);

printf("\nadvance :: %.2f",tech.adv);

tech.dt.m=month;

fwrite(&tech,sizeof(tech),1,t);

fclose(f);

fclose(t);

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

system("del teacher");

system("ren te, teacher");

}

}

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

void fee(int mm)

{

system("cls");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* FEE \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf("\n\t\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

FILE \*f,\*t;

int a=0;

char name[50],c='y';

int clas, roll,month,dif;

while(c=='y'||c=='Y')

{

int a=1,day=0;

fflush(stdin);

printf("\n\nEnter name:: ");

scanf("%[^\n]",name);

printf("\n\nEnter class:: ");

fflush(stdin);

clas=clscanf();

printf("\n\nEnter roll:: ");

fflush(stdin);

scanf("%d",&roll);

f=fopen("student","rb+");

t=fopen("te","wb");

while(fread(&stud,sizeof(stud),1,f)==1)

{

if(strcmp(stud.n,name)==0 && clas==stud.c && roll==stud.r)

{

a=0;

printf("\n\nEnter the month till which fee to be paid:: ");

fflush(stdin);

scanf("%d",&month);

month=chkdat(month,day);

dif=mm-stud.dt.m;

stud.fine=(dif\*stud.f)\*0.01;

stud.due=(dif)\*stud.f;

if (stud.fine<0) stud.fine=0;

if (stud.due<0) stud.due=0;

if (stud.tot<0) stud.tot=0;

stud.tot=stud.fine+stud.due+stud.adv;

printf("\nfine :: %.2f",stud.fine);

printf("\ndue :: %.2f",stud.due);

printf("\ntotal :: %.2f",stud.tot);

printf("\nadvance :: %.2f",stud.adv);

stud.dt.m=month;

stud.tot=0;

stud.fine=0;

stud.due=0;

fwrite(&stud,sizeof(stud),1,t);

}

}

if (a==1)

printf("\n\nRECORD NOT FOUND");

printf("\n\n");

system("pause");

fflush(stdin);

fclose(f);

fclose(t);

system("del student");

system("ren te, student");

printf("\n\nDo you want to continue with the process(press y or Y");

fflush(stdin);

c=getch();

}

getch();

}

void ext()

{

int i;

system("color 0c");

printf("\n\n\t\t Thank you for using C Program School Billing System Project\n\n");

system("pause");

system("cls");

printf("\n\n\t\t\t Exiting\n\n");

for(i=1; i<=80; i++)

{

Sleep(50);

printf("\*");

}

exit(0);

}

int chkdat(int mnt,int dnt)

{

int mon,day;

if (mnt>12 || mnt<1 || dnt<1 || dnt>32)

{

MessageBox(0,"Invalid Date!\nEnter Again","Error!",0);

fflush(stdin);

scanf("%d%d",&mon,&day);

mon=chkdat(mon,day);

}

else

return (mnt);

}

int clscanf()

{

int mnt,mon;

fflush(stdin);

scanf("%d",&mnt);

if (mnt>12 ||mnt<1)

{

MessageBox(0,"Invalid Class!\nEnter Class","Error!!",0);

fflush(stdin);

mon=clscanf();

}

else

return mnt;

}