DELHI TECHNOLOGICAL UNIVERSITY

DATA STRUCTURES (SE-201)

November 17, 2020

TITLE:

BUS RESERVATION SYSTEM

SUBMITTED BY:

MEEDHA (2K19/SE/072)

TO:

Ms. INDU SINGH

LIST OF CONTENTS

ACKNOWLEDGEMENT

I would like to earnestly acknowledge the sincere efforts and valuable time given by my teacher Ms. Indu Singh. Her valuable guidance and feedback has helped me in completing our Data Structures project on the topic Bus Reservation System.

I would also like to thank her for providing us with the required material and guidelines and for encouraging me to complete this project.

Meedha (2K19/SE/072)

ABSTRACT

Traveling is a large growing business across all countries. Bus reservation system deals with maintenance of records of details of each passenger. It also includes maintenance of information like schedule and details of each bus.

We observed the working of the Bus reservation system and after going through it, we got to know that there are many operations, which they have to do manually. It is time-consuming and causes many errors while data entry. Due to this, sometimes a lot of problems occur and they have to resolve disputes with customers. To solve the above problem, and further maintain records of passenger details, seat availability, price per seat, bill generation and other things, we developed a computerized reservation system.

INTRODUCTION

This project, as the name suggests, provides reservation services for buses. This project is designed to perform basic functionalities.

The main purpose for developing this project is to computerize reservation system so that all transactions become fast and to remove the possibility of any errors. It stores all the information of the user in a file using the concepts of file handling.

The user can either be a customer or an admin. A customer and an admin are provided with different services. A customer can view bus details and make reservations as many as he/she wishes and can even search a record to make sure if the reservation has been made or not. The customer can also view the price calculated after making reservation(s). Admin, on the other hand, has the authority to delete, view and modify a record. The admin can count the number of records and can even sort the records.

OBJECTIVES

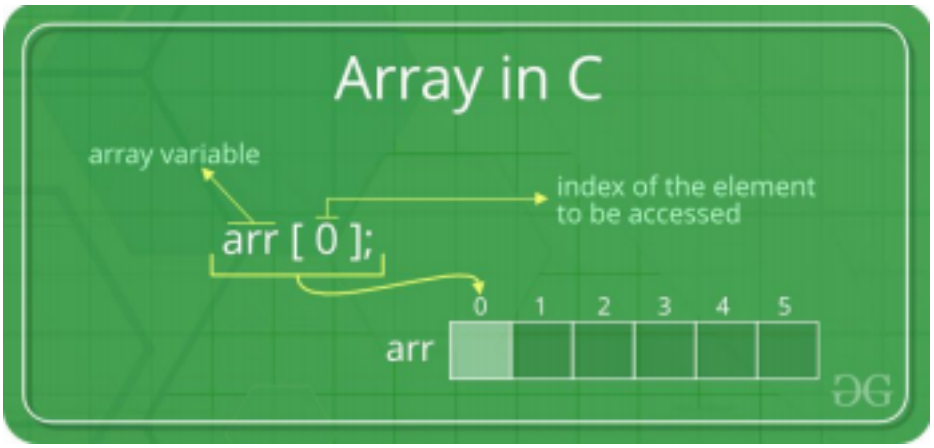
1. **To create a computerized bus reservation system.**
2. **To reduce the possibilities of error while recording data, bill generation etc.**
3. **To create a system where a user can make a bus reservation efficiently.**
4. **To give the user an online experience of the bus reservation system.**

CONCEPTS USED

The data structures concepts used in this project are as follows:

1. Arrays

An array is a collection of items stored at contiguous memory locations. The idea is to store multiple items of the same type together. This makes it easier to calculate the position of each element by simply adding an offset to a base value, i.e., the memory location of the first element of the array (generally denoted by the name of the array). The base value is index 0 and the difference between the two indexes is the offset.



2. File handling

File Handling is the storing of data in a file using a program. In C programming language, the programs store results, and other data of the program to a file using file handling in C. Also, we can extract/fetch data from a file to work with it in the program.

The operations that you can perform on a File in C are –

- Creating a new file
- Opening an existing file
- Reading data from an existing file
- Writing data to a file
- Moving data to a specific location on the file
- Closing the file

We have used the above operations for creating different modules in the project

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

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.

Default open modes

ifstream	ios::in
ofstream	ios::out
fstream	ios::in | ios::out

CODE

```
#include <iostream>
#include <fstream>
#include <conio.h>
#include <stdio.h>
#include <bits/stdc++.h>
#include <string>
#include  <string.h>
#include  <ctype.h>
#include <math.h>

using namespace std;

static int p{0};

void display_detail(int);
void display_codes();
void write();
void read();
void  search_name();
void  delete_record();
void modify();
void sort_name();
void count_res();
```

```
class reservation
9

{
    char first_name[10];
    char last_name[10];
    int age{};
    int date_of_birth{};
    char journey_end[15];
    char journey_start[15];

public:
    int bus1[48],bus2[48],bus3[48],bus4[48],bus5[48];

    reservation()
    {
        age=0;
        date_of_birth=0;
        first_name[0]='\0';
        last_name[0]='\0';
        journey_end[0]='\0';
        journey_start[0]='\0';
    }

    reservation(reservation &R)
    {
        age=R.age;
        date_of_birth=R.date_of_birth;
        strcpy(first_name,R.first_name);
```

```
        strcpy(last_name,R.last_name);
        strcpy(journey_start,R.journey_start);
10
        strcpy(journey_end,R.journey_end);
    }

    void getdata();
    void show();


    char *retfirstname()
    {
        return (first_name);
    }
    char *retlastname()
    {
        return (last_name);
    }
    char *retjs()
    {
        return (journey_start);
    }
    char *retje()
    {
        return (journey_end);
    }

};
```

```
/*-----------------------------------------------------------------------------------------*/
```


```
void reservation :: getdata()
{
    cout<<"\n\t Enter your First name: ";
    cin>>first_name;

    cout<<"\n\t Enter your Last name: ";
    cin>>last_name;

    cout<<"\n\t Enter your Age: ";
    cin>>age;

    cout<<"\n\t Enter your date_of_birth (ddmmyyyy): ";
    cin>>date_of_birth;

    cout<<"\n\t From: ";
    cin>>journey_start;

    cout<<"\n\t To: ";
    cin>>journey_end;

    p++;

}
```

```
/*-----------------------------------------------------------------------------------------*/

void reservation :: show()

12

{
    cout<<" FIRST NAME: "<<first_name<<endl;
    cout<<" LAST NAME: "<<last_name<<endl;
    cout<<" AGE: "<<age<<endl;
    cout<<" DATE OF BIRTH: "<<date_of_birth<<endl;
    cout<<" FROM: "<<journey_start<<endl;
    cout<<" TO: "<<journey_end<<endl;
}

/*-----------------------------------------------------------------------------------------*/

void write()
{
    reservation r;
    ofstream f("details.txt",ios::binary | ios::app );

    char reply;
    do
    {
        r.getdata();
        f.write((char*)&r,sizeof(r));
        cout<<"\n Do you wish to make another reservation? ";
```

```
        cin>>reply;

    }while(reply=='Y' || reply=='y');

    f.close();
}
```


```
/*-----------------------------------------------------------------------------------------*/

void read()
{
    ifstream f("details.txt",ios::binary | ios::in);
    reservation r;

    if(!f)
    {
        cout<<endl<<"\n Error! File does not exist!";
        return;
    }
    int ctr=0;

    while(f.read((char*)&r,sizeof(r)))
    {
        cout<<endl<<"\n Record: "<<++ctr<<endl;
        r.show();
    }
    f.close();
}
```

```
/*-----------------------------------------------------------------------------------------*/

void search_name()
{
   ifstream f("details.txt",ios::in | ios::binary);
14


   char found='N';
   reservation r;

   if(!f)
   {
      cout<<endl<<"\n Error! File does not exist. ";
      return;
   }

   char name[30];

   cin.ignore(numeric_limits<streamsize>::max(),'\n');

   cout<<"\n Enter First name whose record is to be searched: ";
   gets(name);

   while(f.read((char*)&r,sizeof(r)))
   {
      if(strcmp(r.retfirstname(),name)==0)
      {
         cout<<"\n Details of the person are :";
```

```
            r.show();
            found='Y';
            break;
        }
    }
15
    if(found=='N')
    {
        cout<<"\n Sorry! No such record found!"<<endl;
    }

    f.close();
}

/*-------------------------------------------------------------------------------------------*/

void delete_record()
{
    ifstream fmain("details.txt", ios::binary | ios::in);
    ofstream ftemp("temp.dat", ios::binary | ios::out);

    reservation r;

    char name[20];

    cin.ignore(numeric_limits<streamsize>::max(),'\n');
    cout<<"\n Enter First name to delete the record: ";
    gets(name);
```

```
    char found='N';
    while(fmain.read((char*)&r,sizeof(r)))
    {
        if(strcmp(r.retfirstname(),name)!=0)
        {
16
            ftemp.write((char*)&r,sizeof(r));
        }
        else
        {
            found='Y';
        }
    }

    if(found=='N')
    {
        cout<<"\n No such record found!!"<<endl;
    }
    else
    {
        cout<<"\n Record deleted!"<<endl;
    }

    fmain.close();
    ftemp.close();
    remove("details.txt");
    rename("temp.dat","details.txt");
}
```

```
/*-----------------------------------------------------------------------------------------*/

void modify()
{
   fstream f("details.txt", ios::in | ios::out | ios::binary);
17

   reservation r;

   cin.ignore(numeric_limits<streamsize>::max(),'\n');
   char name[20];

   cout<<"\n Enter First name for modification: ";
   gets(name);

   int rec_count=0;
   char found='N';

   while(f.read((char*)&r,sizeof(r)))
   {
      if(strcmp(name,r.retfirstname())==0)
      {
         cout<<"\n Enter New Information";
         r.getdata();

         f.seekg(rec_count*sizeof(reservation),ios::beg);
         f.write((char*)&r,sizeof(reservation));
```

```
            found='Y';
            break;
        }

        rec_count++;
    }
18

    if(found=='Y')
    {
        cout<<"\n Reservation Updated";
    }
    else
    {
        cout<<"\n Record Not Found!!!!";
    }

    f.close();
}

/*-----------------------------------------------------------------------------------------*/

void sort_name()
{
    fstream f;
    f.open("details.txt",ios::binary | ios::in);
    reservation r[500];

    if(!f)
```

```
{
   cout<<"\n Error! File does not exist. ";
   return;
}

int n=0;
while(f.read((char*)&r[n],sizeof(reservation)))
19

 {
    n++;
 }

reservation temp;

int choice;

cout<<"Sort"<<endl;
cout<<"1. Ascending"<<endl;
cout<<"2. Descending"<<endl;
cin>>choice;

if(choice==1)
{
   for(int i=1;i<n;i++)
   {
      for(int j=0;j<n-i;j++)
      {
         if(strcmp(r[j].retfirstname(),r[j+1].retfirstname())>0)
```

```
            {
                temp=r[j];
                r[j]=r[j+1];
                r[j+1]=temp;
            }
        }
    }
}
20

else
{
    for(int i=1;i<n;i++)
    {
        for(int j=0;j<n-i;j++)
        {
            if(strcmp(r[j].retfirstname(),r[j+1].retfirstname())<0)
            {
                temp=r[j];
                r[j]=r[j+1];
                r[j+1]=temp;
            }
        }
    }
}

f.close();
f.open("details.txt",ios::binary | ios::out);

int i=0;
```

```
    while(i<n)
    {
       f.write((char*)&r[i],sizeof(reservation));
       ++i;
    }

    cout<<"\n FILE SORTED"<<endl;
    f.close();
21


}

/*-------------------------------------------------------------------------------------------*/

void count_res()
{
    int num=0;
    reservation r;
    ifstream f("details.txt",ios::in|ios::binary);

    if(!f)
    {
       cout<<"\n Error! File does not exist. "<<endl;
       return;
    }

    while(f.read((char*)&r,sizeof(r)))
    {
       num++;
```

```
    }
    cout<<"\n Number of reservations recorded so far: "<<num;

    f.close();
}

/*-------------------------------------------------------------------------------------------*/
22

int main()
{
    cout<<"\n\n\n\n\n\t\t\t**************************************************************************"<<endl;
    cout<<"\t\t\t\t\tWelcome to the Bus Reservation Portal"<<endl;

    cout<<"\n\n\n\n\n\t\t\t\tCheck the services for admin and customer"<<endl;

    cout<<"\t\t SERVICES FOR CUSTOMER "<<endl;
    cout<<"\t\t\t\t1. Buses codes "<<endl;
    cout<<"\t\t\t\t2. Buses Details "<<endl;
    cout<<"\t\t\t\t3. Get details of the user "<<endl;
    cout<<"\t\t\t\t4. Search a record "<<endl;
    cout<<"\t\t\t\t5. Calculate the total price of the ticket "<<endl;

    cout<<"\t\t SERVICES FOR ADMIN "<<endl;
    cout<<"\t\t\t\t1. Delete a record "<<endl;
    cout<<"\t\t\t\t2. Print details of the user "<<endl;
    cout<<"\t\t\t\t3. Modify a record "<<endl;
    cout<<"\t\t\t\t4. Count the number of records in the file "<<endl;

    cout<<"\t\t\t\t5. Sorting "<<endl;
```

```
cout<<"\n\n\n\n\n\t\t\t**********************************************************************"<<endl;

char ch;
int select;

do{
23


    cout<<"\t\t\t\t CUSTOMER? PRESS 1 "<<endl;
    cout<<"\t\t\t\t ADMIN? PRESS 2 "<<endl;

    cout<<"\t\n Enter your response: ";
    int response{};
    cin>>response;

    if (response==1)
    {
       cout<<"\t\t\t SERVICES FOR CUSTOMER "<<endl;
       cout<<"\t\t\t\t1. Buses codes "<<endl;
       cout<<"\t\t\t\t2. Buses Details "<<endl;
       cout<<"\t\t\t\t3. Get details of the user "<<endl;
       cout<<"\t\t\t\t4. Search a record "<<endl;
       cout<<"\t\t\t\t5. Calculate the total price of the ticket "<<endl;
    }
    else if (response==2)
    {
       cout<<"\t\t\t\t SERVICES FOR ADMIN "<<endl;
       cout<<"\t\t\t\t6. Delete a record "<<endl;
```

```
        cout<<"\t\t\t\t7. Print details of the user "<<endl;
        cout<<"\t\t\t\t8. Modify a record "<<endl;
        cout<<"\t\t\t\t9. Count the number of records in the file "<<endl;
        cout<<"\t\t\t\t10. Sorting"<<endl;
    }

  cout<<"\t\t\t\t\nEnter your choice here: ";
24

  cin>>select;

  switch(select)
  {
    case 1:
    {
        cout<<"\t\n FOLLOWING ARE THE CODES FOR BUSES \n"<<endl;
        display_codes();
        break;
    }

    //*********************************************************
    case 2:
    {
        int number{};
        cout<<"\n Enter the bus number : ";
        cin>>number;
        display_detail(number);
    }
    break;
```

```
//*********************************************************

case 3:
{
   cout<<"\n Enter the details :- \n"<<endl;
   write();
}
break;
25

//*********************************************************

case 4:
   {
      search_name();
   }
break;
//*********************************************************

case 5:
   {
      cout<<"Calculation of total price :"<<" 500 rupees * "<<p<<" = "<<500*p<<endl;
   }
   break;
//*********************************************************

case 6:
   {
      int password{};
      cout<<"\nEnter the password : ";
```

```
            cin>>password;
            if (password==12345)
            {
               delete_record();
            }
            else
            {
               cout<<"\nWrong password!!"<<endl;
26
            }
            break;
         }
      //*************************************************************

      case 7:
         {
            int password{};
            cout<<"Enter the password : ";
            cin>>password;
            if (password==12345)
            {
               read();
            }
            else
            {
               cout<<"\nWrong password!!"<<endl;
            }
            break;
         }
```

```
//*********************************************************

case 8:
  {
     int password{};
     cout<<"Enter the password : ";
     cin>>password;
     if (password==12345)
27

     {
        modify();
     }
     else
     {
        cout<<"\nWrong password!!"<<endl;
     }
  }
break;
//*********************************************************

case 9:
  {
     int password{};
     cout<<"Enter the password : ";
     cin>>password;
     if (password==12345)
     {
        count_res();
     }
```

```
            else
            {
                cout<<"\nWrong password!!"<<endl;
            }
        }
    break;
    //*************************************************************
28

    case 10:
    {
       sort_name();
    }
    break;
    //*************************************************************

    default:
    {
       cout<<"Alert! Invalid entry!"<<endl;
    }
    break;

}

cout<<"\n CONTINUE??";
cin>>ch;
cout<<endl;

}while(ch=='y'||ch=='Y');
```

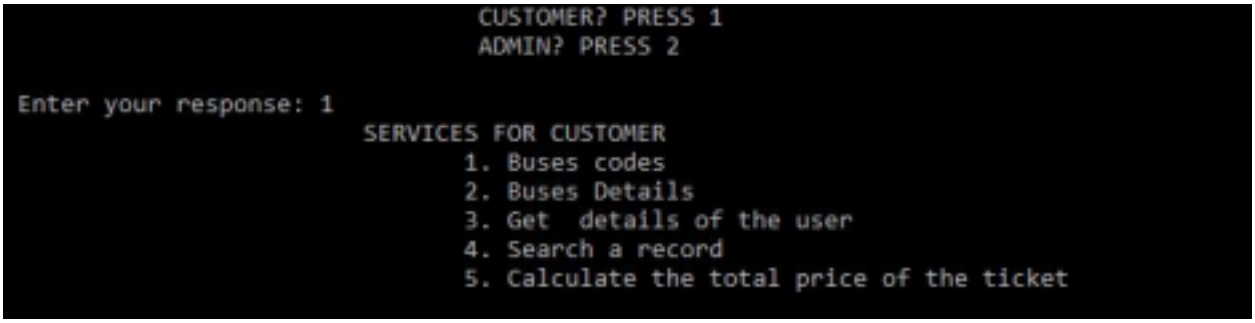
```
    cout<<"\n\n\n\n"<<endl;
    return 0;
}

/*-------------------------------------------------------------------------------------------*/
29

void display_detail(int num)
{
    switch(num)
    {
        case 100:
        {
            cout<<"\n Bus Number: 100"<<endl;
        cout<<" Driver's Name: Mukesh"<<endl;
        cout<<" Number plate: DL 1000 "<<endl; }
        break;

        case 101:
        {
            cout<<"\n Bus Number: 101"<<endl;
        cout<<" Driver's Name: Sanjay"<<endl;
        cout<<" Number plate: DL 1100 "<<endl; }
        break;

        case 102:
        {
```

```
        cout<<"\n Bus Number: 102"<<endl;
        cout<<" Driver's Name: Ali"<<endl;
        cout<<" Number plate: DL 1200 "<<endl;
      }
      break;
30
      case 103:
      {
        cout<<"\n Bus Number: 103"<<endl;
        cout<<" Driver's Name: Ravi"<<endl;
        cout<<" Number plate: DL 1300 "<<endl;
      }
      break;

      case 104:
      {
        cout<<"\n Bus Number: 104"<<endl;
        cout<<" Driver's Name: Rahul"<<endl;
        cout<<" Number plate: DL 1400 "<<endl;
      }
      break;
   }
}

/*-------------------------------------------------------------------------------------------*/

void display_codes()
{
```

```
cout<<" BUS CODES"<<endl;
cout<<"\n Bus Number: 100"<<endl;

cout<<" Journey: "<<endl;
cout<<" From: DELHI TO: AGRA"<<endl;
31

cout<<"\n Bus Number: 101"<<endl;

cout<<" Journey: "<<endl;
cout<<" From: DELHI TO: JAIPUR"<<endl; cout<<"\n Bus

Number: 102"<<endl;

cout<<" Journey: "<<endl;
cout<<" From: DELHI TO: CHANDIGARH"<<endl; cout<<"\n Bus

Number: 103"<<endl;

cout<<" Journey: "<<endl;
cout<<" From: DELHI TO: LUCKNOW"<<endl; cout<<"\n Bus

Number: 104"<<endl;

cout<<" Journey: "<<endl;
cout<<" From: DELHI TO: UDAYPUR"<<endl; }
32
```

OUTPUT

If the user is customer, press1



Services for the customer are displayed

```
Enter your choice here: 1

 FOLLOWING ARE THE CODES FOR BUSES

                        BUS CODES

                   Bus Number: 100
 Journey:
 From: DELHI                                          TO: AGRA

                   Bus Number: 101
 Journey:
 From: DELHI                                          TO: JAIPUR

                   Bus Number: 102
 Journey:
 From: DELHI                                          TO: CHANDIGARH

                   Bus Number: 103
 Journey:
 From: DELHI                                          TO: LUCKNOW

                   Bus Number: 104
 Journey:
 From: DELHI                                          TO: UDAYPUR

 CONTINUE??y
```

Service 1 provides the list of the buses available and their destination

```
                        SERVICES FOR CUSTOMER
                               1. Buses codes
                               2. Buses Details
                               3. Get  details of the user
                               4. Search a record
                               5. Calculate the total price of the ticket

Enter your choice here: 2

 Enter the bus number : 100

 Bus Number: 100
 Driver's Name: Mukesh
 Number plate: DL 1000

 CONTINUE??y
```

```
Enter your choice here: 2

 Enter the bus number : 102

 Bus Number: 102
 Driver's Name: Ali
 Number plate: DL 1200

 CONTINUE??y
```

Service 2 gives further details of that particular bus. The user just needs to enter the bus number for its details.

```
Enter your response: 1
                        SERVICES FOR CUSTOMER
                               1. Buses codes
                               2. Buses Details
                               3. Get  details of the user
                               4. Search a record
                               5. Calculate the total price of the ticket

Enter your choice here: 3

 Enter the details :-


         Enter your First name: Devanshi

         Enter your Last name: Singh

         Enter your Age: 19

         Enter your date_of_birth (ddmmyyyy): 15092001

         From: Delhi

         To: Agra

 Do you wish to make another reservation? y

         Enter your First name: xyz

         Enter your Last name: abc

         Enter your Age: 18

         Enter your date_of_birth (ddmmyyyy): 12092000

         From: Delhi

         To: Chandigarh

 Do you wish to make another reservation? n

 CONTINUE??y
```

Service 3 makes the reservation and saves the personal details of the user into the file permanently.

```
Enter your choice here: 4

Enter First name whose record is to be searched: Devanshi

Details of the person are :                    FIRST NAME:     Devanshi
                  LAST NAME:      Singh
                  AGE:            19
                  DATE OF BIRTH:  15092001
                  FROM:           Delhi
                  TO:             Agra

CONTINUE??
```

Service 4 searches the record

```
Enter your response: 1
                        SERVICES FOR CUSTOMER
                               1. Buses codes
                               2. Buses Details
                               3. Get  details of the user
                               4. Search a record
                               5. Calculate the total price of the ticket

Enter your choice here: 5
Calculation of total price : 500 rupees * 2 = 1000
```

Service 5 gives the price of the ticket

If the user is admin, press 2

```
Enter your response: 2
                              SERVICES FOR ADMIN
                             6. Delete a record
                             7. Print details of the user
                             8. Modify a record
                             9. Count the number of records in the file
                             10. Sorting

Enter your choice here: 6

Enter the password : 12345

 Enter First name to delete the record: Devanshi

 Record deleted!

 CONTINUE??y
```

Service 6 for admin allows the user to delete a record.

```
                              CUSTOMER? PRESS 1
                              ADMIN? PRESS 2

 Enter your response: 2
                              SERVICES FOR ADMIN
                             6. Delete a record
                             7. Print details of the user
                             8. Modify a record
                             9. Count the number of records in the file
                             10. Sorting

Enter your choice here: 7
Enter the password : 12345


 Record: 1
                  FIRST NAME:     xyz
                  LAST NAME:      abc
                  AGE:            18
                  DATE OF BIRTH:  12092000
                  FROM:           Delhi
                  TO:             Chandigarh

 CONTINUE??
```


Service 7 displays the record of files. After deletion there is only 1 record left.

```
Enter your choice here: 8
Enter the password : 12345

 Enter First name for modification: xyz

 Enter New Information
         Enter your First name: abcd

         Enter your Last name: efgh

         Enter your Age: 19

         Enter your date_of_birth (ddmmyyyy): 15092001

         From: Delhi

         To: Agra

 Reservation Updated
 CONTINUE??y
```

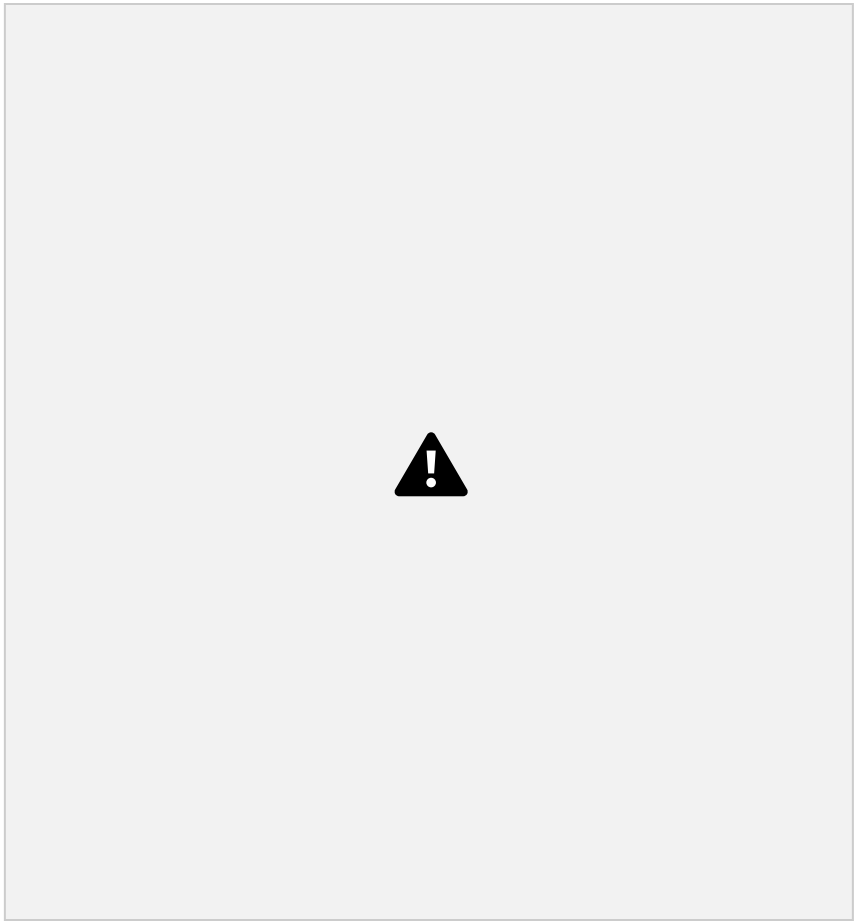
Service 8 modifies a record in the file

```
 Enter your response: 2
                                SERVICES FOR ADMIN
                               6. Delete a record
                               7. Print details of the user
                               8. Modify a record
                               9. Count the number of records in the file
                               10. Sorting

Enter your choice here: 9
Enter the password : 12345

 Number of reservations recorded so far: 1
 CONTINUE??
```

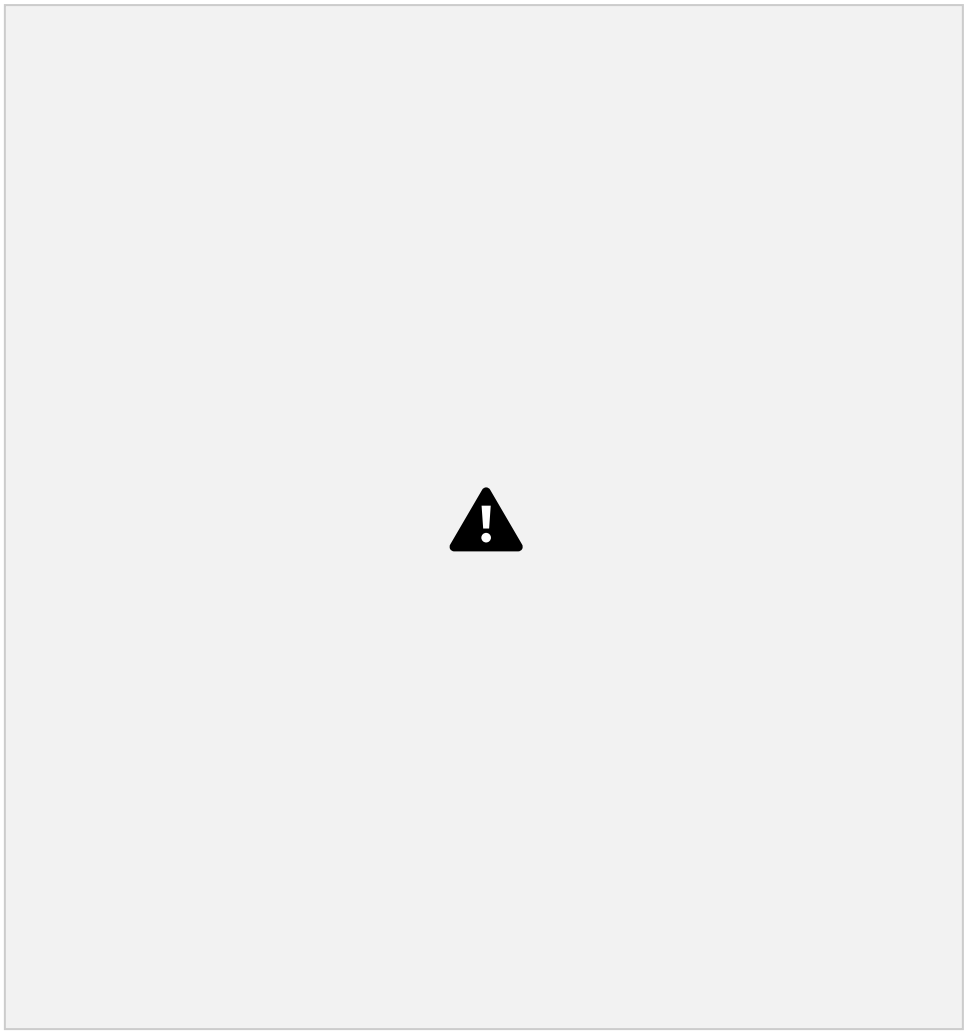
Service 9 shows the number of records in the file

```
                                SERVICES FOR ADMIN
                               6. Delete a record
                               7. Print details of the user
                               8. Modify a record
                               9. Count the number of records in the file
                               10. Sorting

Enter your choice here: 10
Sort
1. Ascending
2. Descending
1

 FILE SORTED

 CONTINUE??y
```

Service 10 sorts the records alphabetically in either ascending or descending order.



In ascending order



In descending order.
40

MODULES

In the beginning, the user is asked whether he/she is a customer or an administrator.

If the user is a customer then following modules are used

1. **MODULE 1**

Buses' codes can be viewed by the user. After viewing the bus code, the user can get further information of that particular bus by entering the bus number such as driver's name, places of arrival and departure.

2. MODULE 2

In this module, the user can make the reservation by calling the service. This means that when the user calls for this service, the user will be asked to fill in their details. After that the user can input the details such as name, age, destination, date of birth, etc. We have used the concept of file handling in this module. It means that as soon as the user will enter the details, they will be stored on a file.

3. MODULE 3

In this module, the users can check details that they had entered when they called for the write data function. The user will be asked to enter their first name. As soon as the user enters their first name, the details of the user will be displayed.

4. MODULE 4

In this module, the user can check the price of the ticket after reserving a number of seats.

If the user is the admin, the user is asked for the password to maintain privacy. The user is then supposed to enter the correct password to use the services of the administrator. The modules are

1. MODULE 1

In this module, when the user calls for this function, the user has to first enter the password and then enter their first name in order to delete a

record. After this, a message will pop up showing that the record was successfully deleted from the database.

2. MODULE 2

In this module, the user can view the records. Again, the user has to enter the correct password after which the user can print all the records recorded so far.

3. MODULE 3

In this module, when the user calls for this function, the user can modify a record if any customer has entered wrong details. The user will be asked for the customer's first name whose data needs to be modified. After that the details can be again filled.

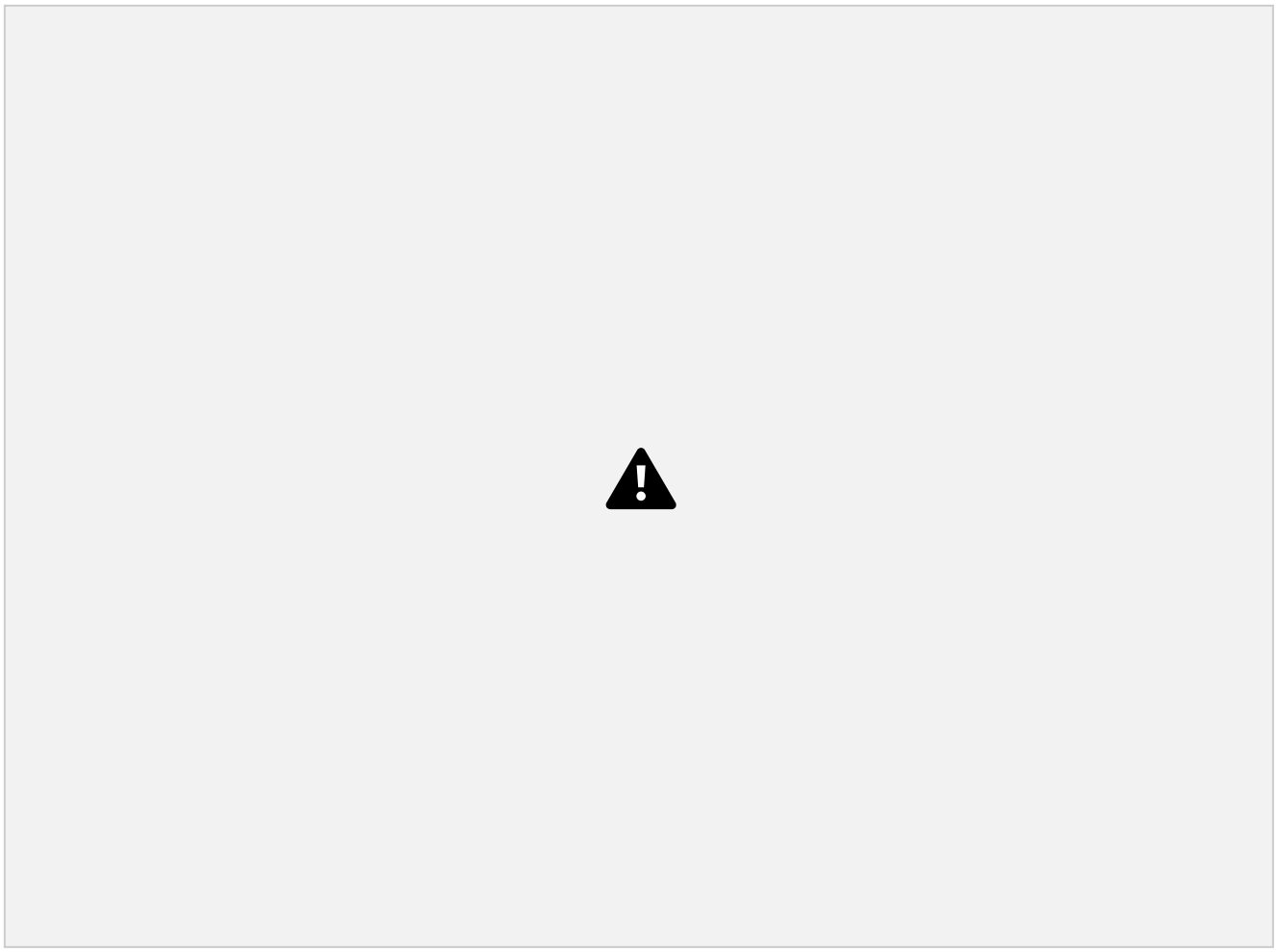
4. MODULE 4

In this module, the user can count the number of records or the reservations that have been made so far.

5. MODULE 5

In this module, the records are sorted in an ascending or descending order using bubble sort as per the wish of the user. Arrays have been used in this module.

FLOWCHART



SYSTEM REQUIREMENTS

IDE used - Code Blocks

Operating system - Any OS

PROGRAMMING LANGUAGE - C++

REFERENCES

https://www.geeksforgeeks.org/

https://www.cplusplus.com/

https://www.edureka.co/blog/file-handling-in-cpp/