

**K LHYDERABAD**  
**FRESHMAN ENGINEERING DEPARTMENT**  
**A Project Based Lab Report**  
**On**  
**PHONEBOOK**

**SUBMITTED BY:**

<u>ID NUMBER</u>	<u>NAME</u>
2110030031	Sumeet Sharma
2110030345	B. Akshaya
2110030037	K. Sushma Reddy
2110030214	Ega Sri Harsha

**UNDER THE ESTEEMED GUIDANCE OF**  
**Dr. Ramasubramanian Krishnamurthy**  
**Associtive Professor**



**KONERU LAKSHMAIAH EDUCATION FOUNDATION**

(Deemed to be University)  
Moinabad Road, Aziz Nagar, Hyderabad - 500075

## DEPARTMENT OF BASIC ENGINEERING SCIENCES

### CERTIFICATE

This is to certify that the project based laboratory report entitled "PHONEBOOK" submitted by Mr. Sumeet Sharma(2110030031) , Ms. B. Akshaya(2110030345) , Ms. K. Sushma Reddy(2110030037). Ms. Ega Sri Harsha(2110030214) to the **Department of Basic Engineering Sciences, KL University** in partial fulfillment of the requirements for the completion of a project in "Data Structures-21SC1202" course in I B Tech II Semester, is a bonafide record of the work carried out by him/her under my supervision during the academic year 2021-22.

PROJECT SUPERVISOR

HEAD OF THE DEPARTMENT

Dr. Ramasubramanian Krishnamurthy

Dr. G. REKHA

## ACKNOWLEDGEMENTS

It is great pleasure for me to express my gratitude to our honorable President **Sri. Koneru Satyanarayana**, for giving the opportunity and platform with facilities in accomplishing the project based laboratory report.

I express the sincere gratitude to our Principal **Dr.L. Koteswara Rao** for his administration towards our academic growth.

I express sincere gratitude to our Coordinator **Dr. G. Rekha** for her leadership and constant motivation provided in successful completion of our academic semester. I record it as my privilege to deeply thank for providing us the efficient faculty and facilities to make our ideas into reality.

I express my sincere thanks to our project supervisor **Dr. Ramasubramanian Krishnamurthy** for her novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

## **ABSTRACT**

This project is about Phonebook in data structures.

The main aim of this project is creating contacts (new contacts) and operating an Phonebook. It is where we can save contacts of the people we know. We can create contacts with the details like name, phone number, father name, address, email, citizenship number for every contact and save it in our phonebook. We can modify the details of a particular contact whenever we need and can also delete a particular contact. There is an option called search to search the contact instead of checking the whole phonebook one-by-one. We can search the contact by giving any of the one information like name, phone number, father name, address, email or citizenship number.

We can learn about Phonebook in data structures in this project.

## INDEX

S.NO	TITLE	PAGE NO
1	Introduction	<6>
2	Aim of the Project	<6>
2.1	Advantages	<6>
3	Software & Hardware Details	<7>
4	Flow charts	<8>
5	Algorithm	<9-10>
6	Implementation, code	<11-41>
7	Output screenshots	<42>
8	Conclusion	<43>

# INTRODUCTION

The main aim of this project is creating contacts (new contacts) and operating an Phonebook. It is where we can save contacts of the people we know. We can create contacts with the details like name, phone number, father name, address, email, citizenship number for every contact and save it in our phonebook. We can modify the details of a particular contact whenever we need and can also delete a particular contact. There is an option called search to search the contact instead of checking the whole phonebook one-by-one. We can search the contact by giving any of the one information like name, phone number, father name, address, email or citizenship number.

**Requirements:** To implement this project student should have knowledge on

1. Structures.
2. Switch case statements, if-else, looping statements.
3. Files concept.

## AIM

THE MAIN AIM OF THIS PROJECT IS CREATING AND OPERATING A PHONEBOOK.

### Advantages:-

- Can maintain the list of information of a person.
- Can use it whenever required as it is stored or saved and doesn't require us to remember everything.
- Can modify, search or delete.

# SYSTEM REQUIREMENTS

## ➤ SOFTWARE REQUIREMENTS:

The major software requirements of the project are as follows:

Language : C

Operating system: Windows XP or later.

Tools: DEV C++.

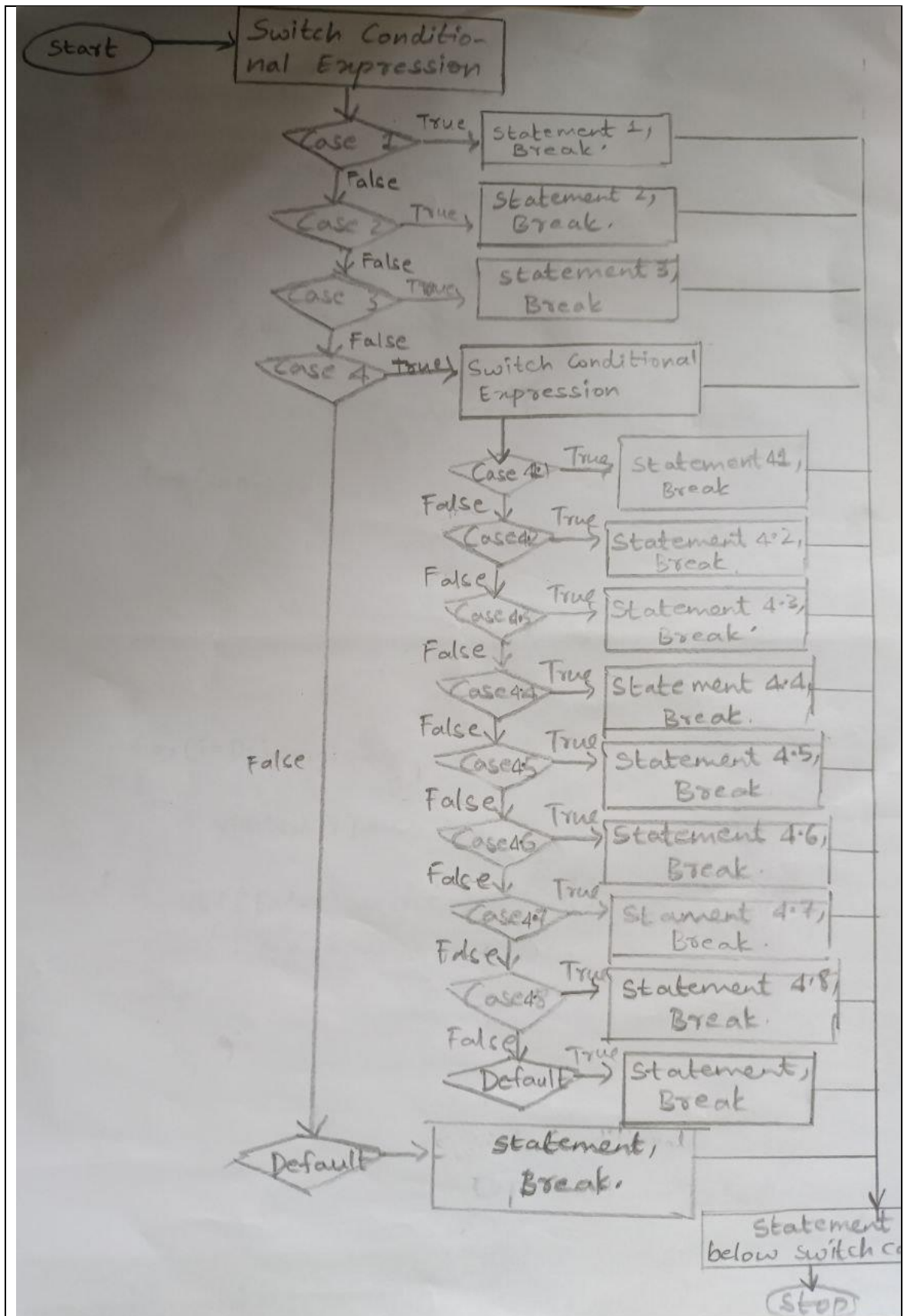
## ➤ HARDWARE REQUIREMENTS:

The hardware requirements that map towards the software are as follows:

RAM : 4 GB Min

Processor : i3 10<sup>rd</sup> Gen

# FLOW CHART





## ALGORITHM

○ Step1: Start.

○ Step2: It shows 6 option to select,

1.Create contact.

2.Display contact.

3.Modify contact.

4.Search contact.

4.1. Search by name.

4.2. Search by phone number.

4.3. Search by gender.

4.4. Search by father name.

4.5. Search by phone number.

4.6. Search by citizenship number.

4.7. Search by email.

4.8. Search by address.

5.Delete contact.

○ Step3: If the selection number is less than or equal 5, it shows its output else shows error.

○ Step4: If the selected number is 1, we should create the contact by entering name, phone number, gender,

father name, citizenship number, email and address.  
The information gets saved successfully.

If the selection is 2 it displays the saved contacts.

If the selection is 3, it asks to enter contact name to modify, if found enter details again with the modifications else it displays “person not found” message.

If the selection is 4, we can search for a particular contact in different ways as given above, we need to select a number,

If the selection number is less than or equal 7, it shows its output else it prints error message.

If the selection is 5, it deletes the contact by taking contact name and

If the selection is 6, we exit.

○ Step5: This will repeat again and again until we exit.

○ Step6: Stop.

# IMPLEMENTATION

## CODE

```
#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<windows.h>

#include<conio.h>

struct phone_book

{ char name[30];

    char sex;

    char Fathername[30];

    char phone_number[20];

    char citizenship_number[20];

    char email[30];

    char address[30];

};

typedef struct phone_book pb;

pb p[100];

int test;

void Create(int i);

void DisplayAll();

void Modify();

void Delete();

void Search_by_Name();

void Search_by_Gender();

void Search_by_FatherName();
```

```
void Search_by_phoneNumber();  
void Search_by_citizenship_number();  
void Search_by_Address();  
void Search_by_Email();  
void sort();  
void gotoxy(int,int);  
void boxout(int l,int b,int x,int y);
```

```
void boxout(int l,int b,int x,int y)  
{  
    int i;  
    for(i=0;i<l;i=i+2)  
    {  
  
        gotoxy(x+i,y);  
        printf("_");  
        gotoxy(x+i,y+b);  
        printf("_");  
    }  
    for(i=0;i<b;i++)  
    {  
        gotoxy(x,y+i);  
        printf("|");  
        gotoxy(x+l,y+i);  
        printf("|");  
    }  
    gotoxy(x+l,y);
```

```

        printf(" ");
        gotoxy(x+l,y+b);
        printf("|");
        gotoxy(x,y);
        printf(" ");
        gotoxy(x,y+b);
        printf("|");
    }

void linebox(int l,int b,int x,int y)
{
    int i;
    for(i=0;i<l;i++)
    {
        gotoxy(x+i,y);
        printf("%c",196);
        gotoxy(x+i,y+b);
        printf("%c",196);
    }
    for(i=0;i<b;i++)
    {
        gotoxy(x,y+i);
        printf("%c",179);
        gotoxy(x+l,y+i);
        printf("%c",179);
    }

    gotoxy(x+l,y);
    printf("%c",191);
    gotoxy(x+l,y+b);

```

```

        printf("%c",217);

        gotoxy(x,y);
        printf("%c",218);
        gotoxy(x,y+b);
        printf("%c",192);
    }
    void gotoxy(int x , int y)
    {
        COORD c;

        c.X=x;
        c.Y=y;
        SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),c);
    }


    int main()
    {
        int ch,x,y;

        static int i=0;

        int k;

        FILE *fp;


        linebox(42,17,90,2);
        boxout(42,17,90,2);

        gotoxy(91,5);

        printf("\tWELCOME TO PHONEBOOK PROJECT ");

        gotoxy(91,6);

```

```

printf("-----");

gotoxy(91,7);

printf("\t\tTEAM MEMBERS");

gotoxy(91,8);

printf("-----");

gotoxy(91,9);

printf("2110030031 - Sumeet Sharma");

gotoxy(91,11);

printf("2110030345 - B.Akshaya");

gotoxy(91,13);

printf("2110030037 - Sushma Reddy");

gotoxy(91,15);

printf("210030214 - Sri Harsha");

gotoxy(91,17);

printf("-----");


linebox(35,3,1,50);

boxout(35,3,1,50);

gotoxy(2,52);

printf("Enter any Number to Continue :");

scanf(" %d",&k);

if(k==1)

system("cls");

else

system("cls");


do{

linebox(28,12,90,2);

```

```

        boxout(28,12,90,2);

        gotoxy(91,5);

        printf("Enter");

        gotoxy(91,6);

        printf("1.Create Contact");

        gotoxy(91,7);

        printf("2.Display List of Contacts");

        gotoxy(91,8);

        printf("3.Modify Contact");

        gotoxy(91,9);

        printf("4.Search Contact");

        gotoxy(91,10);

        printf("5.Delete Contact");

        gotoxy(91,11);

        printf("6.Exit");

        gotoxy(91,12);

        scanf("%d",&ch);

        system("cls");

        switch(ch)

        {

        case 1: fp=fopen("Record.txt","a");

                test=0;

                linebox(80,30,90,2);

                boxout(80,30,90,2);

                Create(i);

                if(test==0)

        {fwrite(&p[i],sizeof(pb),1,fp);

        //Writing the Data in the file

```



```
        if(fwrite!=0)
            printf("Written Successfully !!\n");
        else
            printf("Error in Writing File\n");
        i++;
    printf("Contact Created Successfully !!!");
    }
    fclose(fp);
    break;
case 2: sort();
    DisplayAll();
    break;
case 3: Modify();
    break;
case 4:
    do{
        linebox(32,15,90,2);
        boxout(32,15,90,2);
        gotoxy(91,5);
        printf("Enter");
        gotoxy(91,6);
        printf("1.Search by Name");
        gotoxy(91,7);
        printf("2. Search by Gender");
        gotoxy(91,8);
        printf("3. Search by Father Name");
        gotoxy(91,9);
        printf("4.Search by Phone Number");
```

```

gotoxy(91,10);

printf("5.Search by Citizenship Number");

gotoxy(91,11);

printf("6. Search by Email");

gotoxy(91,12);

printf("7.Serch by Address");

gotoxy(91,13);

printf("8. Exit");

gotoxy(91,14);

scanf("%d",&y);

gotoxy(1,1);

switch(y)

{ case 1:Search_by_Name();break;

case 2:Search_by_Gender();break;

case 3:Search_by_FatherName();break;

case 4:Search_by_phoneNumber();break;

case 5:Search_by_citizenship_number();break;

case 6:Search_by_Email();break;

case 7:Search_by_Address();break;


default: if(y>8)

printf("Invalid Choice !!");

}

linebox(35,3,1,50);

boxout(35,3,1,50);

gotoxy(2,52);

printf("Enter any Number to Continue :");

scanf(" %d",&k);

```

```
        if(k==1)
            system("cls");
        else
            system("cls");
    }while(y!=8);

    break;

    case 5: Delete();
    break;

    case 6: exit(1);
}    linebox(35,3,1,50);
    boxout(35,3,1,50);
    gotoxy(2,52);
printf("Enter any Number to Continue :");
scanf(" %d",&k);
    if(k==1)
        system("cls");
    else
        system("cls");

}while(1);

}
```

```

int Verify_by_Name(char s1[])                                     //Searching for
    Contact By Name if it alredy exists

    {

        int f=0;

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

        while(fread(&s, sizeof(pb), 1, op))
        { if(strcmp(s1,s.name)==0)
            f=1;
        }

        fclose (op);

        return f;

    }

```

```

int Verify_by_phoneNumber(char s1[])                             // Searching for Contact from
    Contact List By PhoneNumber

    {

        int f=0;

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

        while(fread(&s, sizeof(pb), 1, op))
        { if(strcmp(s.phone_number,s1)==0)
            {
                f=1;
            }
        }
    }

```

```

    }

    }

    fclose (op);

    return f;

    }

void Create(int i)

    //Creating Contact in Contact List

    { int x,j,n,k=5;

      char b[12];

      gotoxy(91,k++);

      printf("Enter the Persons Details:");

      gotoxy(91,k++);

      printf("Name :");

      scanf(" %[^\n]s",p[i].name);

      if(Verify_by_Name(p[i].name)) // Checking name in
          ContactList

          {      gotoxy(91,k++);

                printf("__Contact Alredy Exists with this Name__");

                test=1;

                }

          else

          {

              do

              { x=0;

                gotoxy(91,k++);

```

```

        printf("Phone Number :");

scanf(" %s",p[i].phone_number);

n=strlen(p[i].phone_number);

if(n!=10)

    {x=1;

        gotoxy(91,k++);

printf("__Enter 10 Digits Number!!__");

        }

    else

        {

for(j=0;j<strlen(p[i].phone_number);j++)

            {x=0;

                b[j]=p[i].phone_number[j];

                    if(isdigit(b[j]))

                        {

                            }

                                else

                                    {

                                        gotoxy(91,k++);

printf("__Enter only Numbers__");

                                            x=1;

                                                break;

                                                    }

                                                        }

                                                            }

                                                                if(x==0)

```

```
if(Verify_by_phoneNumber(p[i].phone_number))           //Checking Phone
Number in Contactlist
```

```
{gotoxy(91,k++);
```

```
printf("__Contact Alredy Exists with this Phone Number__");
```

```
gotoxy(91,k++);
```

```
printf("--Do you want to Continue--  Enter 1 to continue 0 to stop");
```

```
gotoxy(91,k++);
```

```
scanf("%d",&x);
```

```
if(x==0)
```

```
test=1;
```

```
}
```

```
}while(x!=0);
```

```
if(test!=1)                                           //Validation for M or F
```

```
{
```

```
do
```

```
{gotoxy(91,k++);
```

```
printf("Enter M or F");
```

```
gotoxy(91,k++);
```

```
printf("Sex :");
```

```
scanf(" %c",&p[i].sex);
```

```
if(p[i].sex=='M' || p[i].sex=='F')
```

```
x=0;
```

```
else
```

```
x=1;
```

```
}while(x==1);
```

```

                                gotoxy(91,k++);

printf("Fathername :");           //Accepting Father Name

scanf(" %[^\\n]s",p[i].Fathername);

                                do{x=0;

                                gotoxy(91,k++);

                                printf("Citizenship Number :");

scanf(" %s",p[i].citizenship_number);

                                n=strlen(p[i].citizenship_number);

if(n!=12)                        //Validation to Enter 12 Digits

                                {x=1;

                                gotoxy(91,k++);

                                printf("__Enter 12 Digits Number!!__");

                                }

                                else

                                {

                                for(j=0;j<strlen(p[i].citizenship_number);j++)

                                {x=0;

                                b[j]=p[i].citizenship_number[j];

if(isdigit(b[j]))                //Validation to Enter only Number in

                                Citizenship Number

                                {

                                }

                                else

                                {gotoxy(91,k++);

                                printf("__Enter only Numbers__");

                                x=1;

                                break;

```



```

        }

    }}

    }while(x!=0);

    gotoxy(91,k++);

    printf("Email :");                //Accepting Email
    scanf(" %s",p[i].email);

    gotoxy(91,k++);

    printf("Address :");                //Accepting Address
    scanf(" %[^\\n]s",p[i].address);

    }

    }

    }

void DisplayAll()
    // Displaying All the Conatcts in the Contact List

    { pb s;

    FILE *op;

    op=fopen("Record.txt","r");

    while(fread(&s, sizeof(pb), 1, op))                //Reading the
        Data From the file

        {

            printf("\\nName :\\t%s",s.name);

            printf("\\nPhone Number :\\t%s",s.phone_number);

            printf("\\nSex :\\t%c",s.sex);

```

```

        printf("\nFathername :\t%s",s.Fathername);

        printf("\nCitizenship Number :\t%s",s.citizenship_number);

        printf("\nEmail :\t%s",s.email);

        printf("\nAddress :\t%s",s.address);

        printf("\n\n");

        }

        fclose (op);

    }

void Search_by_Name()

    //Searching for Contact By Name

    { char s1[30];

        int f=0;

        printf("Enter the Person to Search\nName :");

        scanf(" %[^\\n]s",s1);

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

        while(fread(&s, sizeof(pb), 1, op))

        { if(strcmp(s1,s.name)==0)

            {

                printf("\nName :\t%s",s.name);

                printf("\nPhone Number :\t%s",s.phone_number);

                printf("\nSex :\t%c",s.sex);

                printf("\nFathername :\t%s",s.Fathername);

                printf("\nCitizenship Number :\t%s",s.citizenship_number);

                printf("\nEmail :\t%s",s.email);

```

```

        printf("\nAddress : \t%s",s.address);

        printf("\n\n");

        f=1;

        }

    }

    if(f==0)

printf("\n__Person Not Found in Contacts__\n");

        fclose (op);

    }

```

**void Modify()**

//

**Modifying Contact in the Conatct List**

```

        { char s1[30],b[20];

        int f=0,x,n,j,ch;

printf("Enter the Contact Name to Modify");

        scanf(" %[^\\n]s",s1);

        pb s;

        FILE *op,*op1;

        op=fopen("Record.txt","r");

        op1=fopen("Temp.txt","w");

        while(fread(&s, sizeof(pb), 1, op))

        { if(strcmp(s1,s.name)==0)

        { printf("\nName : \t%s",s.name);

printf("\nPhone Number : \t%s",s.phone_number);

        printf("\nSex : \t%c",s.sex);

```

```

printf("\nFathername :\t%s",s.Fathername);

printf("\nCitizenship_number :\t%s",s.citizenship_number);

printf("\nEmail :\t%s",s.email);

printf("\nAddress :\t%s",s.address);

printf("\n\n");


do{

printf("Enter\n1.To Modify Name\n2.To Modify PhoneNumber\n3. To modify
Gender\n4.To Modify FatherName\n5.To Modify CiteZenship Number\n6.To Modify
Email \n7.To Modify Address\n8.Save and Exit");

scanf("%d",&ch);

fflush(stdin);

printf("\nEnter the Persons Details: \n");

switch(ch)

{

case 1:

printf("Name :");

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

break;

case 2:

do

{ x=0;

printf("Phone Number :");

scanf(" %s",s.phone_number);

n=strlen(s.phone_number);

if(n!=10)

{x=1;

```

```

printf("\n_Enter 10 Digits Number!!_\n");

    }

    else

    {

for(j=0;j<strlen(s.phone_number);j++)

    {x=0;

    b[j]=s.phone_number[j];

    if(isdigit(b[j]))

    {

    }

    else

    {

printf("\n_Enter only Numbers_\n");

        x=1;

        break;

    }

    }

    }

}while(x!=0);

break;

case 3:

do

    { x=0;

printf("\nSex :");

scanf(" %c",&s.sex);

if(s.sex=='M' || s.sex=='F')

    {

```

```

        x=0;

        }

        else

        {printf("\n_Enter M or F!!_\n");

        x=1;

        }

    }while(x==1);

    break;

    case 4:

    printf("\nFathername :");

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

    break;

    case 5:

    do

    {

    printf("\nCitizenship Number :");

    scanf(" %s",s.citizenship_number);

    n=strlen(s.citizenship_number);

    if(n!=12)

    {printf("\nEnter 12 Digits Number !!_\n");

    x=1;

    }

    else

    {

    for(j=0;j<n;j++)

    {x=0;

    b[j]=s.citizenship_number[j];

```

```

if(isdigit(b[j])) //Validation to Enter only Number in
                  Citizenship Number
    {
    }
    else
    {
    printf("\n_Enter only Numbers_\n");
    x=1;
    break;
    }
    }}
}while(x==1);
    break;
    case 6:
    printf("\nEmail :");
    scanf(" %s",s.email);
    break;
    case 7:
    printf("\nAddress :");
    scanf(" %[^\n]s",s.address);
    f=1;
    break;
    case 8: exit(1);
    }
}while(ch<=8);
    }
fwrite(&s,sizeof(pb),1,op1);
    }

```

```

        fclose(op);

        fclose(op1);

        if(f==1)
        {op1=fopen("Temp.txt","r");
        op=fopen("Record.txt","w");

        while(fread(&s,sizeof(pb),1,op1))
        {fwrite(&s,sizeof(pb),1,op);

        }

        fclose(op);
        fclose(op1);
        printf("Contact Modified Successfully !!!");
        }
        else
        printf("\n__Person Not Found in Contacts__\n");
        }

```

```

void Delete()                                // Deleting Contact From Conatct List
{ char s1[30];
  int f=0;
  printf("Enter the Contact Name to Delete\nName:");
  scanf(" %[^\\n]s",s1);
  pb s;
  FILE *op,*op1;
  op=fopen("Record.txt","r");

```



```

        op1=fopen("Temp.txt","w");
        while(fread(&s, sizeof(pb), 1, op))
        { if(strcmp(s1,s.name)==0)
            {
                f=1;
            }
            else
            {
                fwrite(&s,sizeof(pb),1,op1);
            }
        }
        fclose(op);
        fclose(op1);
        if(f==1)
        {op1=fopen("Temp.txt","r");
        op=fopen("Record.txt","w");

        while(fread(&s,sizeof(pb),1,op1))
        {fwrite(&s,sizeof(pb),1,op);

            }

            fclose(op);
            fclose(op1);
            printf("Contact Deleted Successfully !!!");
        }
        else
        printf("\n__Person Not Found in Contacts__\n");

```

```

    }

void Search_by_Gender() //
    Searching Contact from Contact List By Gender

    { char s1;

    printf("Enter the M (Male) or F (Female) \nsex :\n");

    scanf(" %c",&s1);

    int f=0;

    pb s;

    FILE *op;

    op=fopen("Record.txt","r");

    while(fread(&s, sizeof(pb), 1, op))
    { if(s.sex==s1)
        {
            printf("\nName :\t%s",s.name);
            printf("\nPhone Number :\t%s",s.phone_number);
            printf("\nSex :\t%c",s.sex);
            printf("\nFathername :\t%s",s.Fathername);
            printf("\nCitizenship Number :\t%s",s.citizenship_number);
            printf("\nEmail :\t%s",s.email);
            printf("\nAddress :\t%s",s.address);

            printf("\n\n");

            f=1;
        }
    }

```

```

        }

        if(f==0)

printf("\n__Person Not Found in Contacts__\n");

        fclose (op);

    }

```

```

void Search_by_FatherName()                                // Searching For
    Contact from Contact List By Father Name

    { char s1[30];

printf("Enter Father's Name \nFathername :");

    scanf(" %[^\n]s",s1);

    int f=0;

    pb s;

    FILE *op;

    op=fopen("Record.txt","r");

    while(fread(&s, sizeof(pb), 1, op))

    { if(strcmp(s1,s.Fathername)==0)

        {

            printf("\nName :\t%s",s.name);

printf("\nPhone Number :\t%s",s.phone_number);

            printf("\nSex :\t%c",s.sex);

printf("\nFathername :\t%s",s.Fathername);

printf("\nCitizenship Number :\t%s",s.citizenship_number);

            printf("\nEmail :\t%s",s.email);

printf("\nAddress :\t%s",s.address);

```

```

        printf("\n\n");

        f=1;

        }

    }

    if(f==0)

printf("\n__Person Not Found in Contacts__\n");

    fclose (op);

    }

void Search_by_phoneNumber()                                // Searching for Contact
    from Contact List By PhoneNumber

    {

        char s1[20];

printf("Enter the Phone Number \nPhone Number :");

        scanf(" %s",&s1);

        int f=0;

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

        while(fread(&s, sizeof(pb), 1, op))

        { if(strcmp(s.phone_number,s1)==0)

            {

                printf("\nName :\t%s",s.name);

printf("\nPhone Number :\t%s",s.phone_number);

                printf("\nSex :\t%c",s.sex);

printf("\nFathername :\t%s",s.Fathername);

```

```

printf("\nCitizenship Number :\t%s",s.citizenship_number);

printf("\nEmail :\t%s",s.email);

printf("\nAddress :\t%s",s.address);

printf("\n\n");

f=1;

}

}

if(f==0)

printf("\n__Person Not Found in Contacts__\n");

fclose (op);

}

void Search_by_citizenship_number() // Searching for
Contact from Contact List by Citizenship Number

{

char s1[20];

printf("Enter the Citizenship Number \ncitizenship_number :");

scanf(" %s",&s1);

int f=0;

pb s;

FILE *op;

op=fopen("Record.txt","r");

while(fread(&s, sizeof(pb), 1, op))

{ if(strcmp(s.citizenship_number,s1)==0)

{

printf("\nName :\t%s",s.name);

```

```

        printf("\nPhone Number :\t%s",s.phone_number);

        printf("\nSex :\t%c",s.sex);

        printf("\nFathername :\t%s",s.Fathername);

        printf("\nCitizenship Number :\t%s",s.citizenship_number);

        printf("\nEmail :\t%s",s.email);

        printf("\nAddress :\t%s",s.address);

        printf("\n\n");

        f=1;

        }

    }

    if(f==0)

        printf("\n__Person Not Found in Contacts__\n");

        fclose (op);

    }

void Search_by_Address()
//Searching for Contact from Contact List By Address

    { char s1[30];

        printf("Enter the Address ");

        scanf(" %[^\\n]s",&s1);

        int f=0;

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

        while(fread(&s, sizeof(pb), 1, op))

        { if(strcmp(s1,s.address)==0)

```

```

        {

            printf("\nName :\t%s",s.name);

            printf("\nPhone Number :\t%s",s.phone_number);

            printf("\nSex :\t%c",s.sex);

            printf("\nFathername :\t%s",s.Fathername);

            printf("\nCitizenship Number :\t%s",s.citizenship_number);

            printf("\nEmail :\t%s",s.email);

            printf("\nAddress :\t%s",s.address);

            printf("\n\n");

            f=1;

        }

    }

    if(f==0)

        printf("\n__Person Not Found in Contacts__\n");

        fclose (op);

    }

```

```

void Search_by_Email() //
    Searching for Contact from Contact List By Email

    { char s1[30];

        printf("Enter the Email ");

        scanf(" %[^\\n]s",&s1);

        int f=0;

        pb s;

        FILE *op;

        op=fopen("Record.txt","r");

```

```

while(fread(&s, sizeof(pb), 1, op))
{ if(strcmp(s1,s.email)==0)
{
printf("\nName :\t%s",s.name);
printf("\nPhone Number :\t%s",s.phone_number);
printf("\nSex :\t%c",s.sex);
printf("\nFathername :\t%s",s.Fathername);
printf("\nCitizenship Number :\t%s",s.citizenship_number);
printf("\nEmail :\t%s",s.email);
printf("\nAddress :\t%s",s.address);
printf("\n\n");
f=1;
}

}

if(f==0)
printf("\n__Person Not Found in Contacts__\n");
fclose (op);
}

```

**void sort()**

**// Code for Sorting Data in File**

```
{ int i,j;
```

```
pb *s,s1;
```

```
FILE *fp;
```

```
fp=fopen("Record.txt","r");
```

```
fseek(fp,0,SEEK_END);
```

**//fseek() used to move file pointer associated with a given file to a specific position.**



```

int n=ftell(fp)/sizeof(pb);                                //ftell() returns the current file position

s=(pb*)calloc(n,sizeof(pb));                                //used to allocate a specified amount of
memory and then initialize it to zero

rewind(fp);                                                //sets the file position to the beginning of the file

for(i=0;i<n;i++)
fread(&s[i],sizeof(pb),1,fp);
fclose(fp);

for(i=1;i<n;i++)
{for(j=0;j<n-i;j++)
{
if(strcmp(s[j].name,s[j+1].name)>0)
{
    s1=s[j];
    s[j]=s[j+1];
    s[j+1]=s1;
}
}
}

fp=fopen("Record.txt","w");

for(i=0;i<n;i++)
{
fwrite(&s[i],sizeof(pb),1,fp);
}
fclose(fp);
}

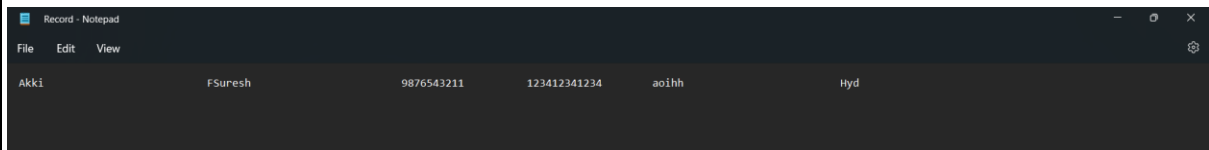
```

# EXECUTION:

```
C:\Users\pc\Desktop\EVEN SEM\DS\PROJECT\ProjectFinal.exe
Enter
1.Create Contact
2.Display List of Contacts
3.Modify Contact
4.Search Contact
5.Delete Contactt
6.Exit
-
```

```
C:\Users\pc\Desktop\EVEN SEM\DS\PROJECT\ProjectFinal.exe
Name : Akki
Phone Number : 9876543210
sex : F
Fathername : Suresh
citizenship_number : 123412341234
email : akki@gmail.com
address : Hyderabad

Name : Akshaya
Phone Number : 9753186542
sex : F
Fathername : Mahesh
citizenship_number : 432143214321
email : aks@gmail.com
address : Hyd
```



## CONCLUSION

### Future Work

This project is about Phonebook in data structures.

The main aim of this project is creating contacts (new contacts) and operating an Phonebook. It is where we can save contacts of the people we know. We can create contacts with the details like name, phone number, father name, address, email, citizenship number for every contact and save it in our phonebook. We can modify the details of a particular contact whenever we need and can also delete a particular contact. There is an option called search to search the contact instead of checking the whole phonebook one-by-one. We can search the contact by giving any of the one information like name, phone number, father name, address, email or citizenship number.