KIET Group of Institutions, Ghaziabad

COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

PROJECT BASED LEARNING

on

PHONEBOOK MANAGEMENT SYSTEM

SUBJECT: DATA STRUCTURES USING C LAB

(KCS-301)

Submitted By:

RUDRANSH SRIVASTAVA	**- 2100290110135**
UTSAH SHANKER	**- 2100290110178**
VANSH SHARMA	**- 2100290110185**

ACKNOWLEDGEMENT

I've got this golden opportunity to express my kind gratitude and sincere thanks to my subject faculty **"Mr. Vinay Kumar"**, Computer Science and Information Technology Department, **KIET GROUP OF INSTITUTIONS** fortheir kind support and necessary counselling in the preparation of this project report. I'm also indebted to every person responsible for the making up of this project directly or indirectly.

I must also acknowledge or deep debt of gratitude each one of my colleagues who led this project come out in the way it is. It's my hard work and untiring sincere efforts and cooperation to bring out the project work. Last but not the least, I would like to thank my parents for their sound counselling and cheerful support. They have always inspired us and kept our spirit up.

ABSTRACT

The primary features that make up the main menu of this Phonebook program include adding new records, listing them, altering and updating them, searching for stored contacts, and deleting phonebook data.

When adding a record to the Phonebook, personal information such as name, gender, father's name, phone number, citizenship number, email, and address is requested. Modifying, listing, searching for, and removing these records is then possible.

INDEX

<u>INTRODUCTION</u>

<u>Problem Statement:</u> To make a phonebook management system. To use to linked list, array data structures to achieve the aim.

<u>Purpose:</u> The primary features that make up the main menu of this Phonebook program include adding new records, listing them, altering and updating them, searching for stored contacts, and deleting phonebook data.

REQUIREMENT ANALYSIS

Functional Requirements: No additional functional requirements required.

Dependencies: The program is made in C language so it will run on a C compiler.
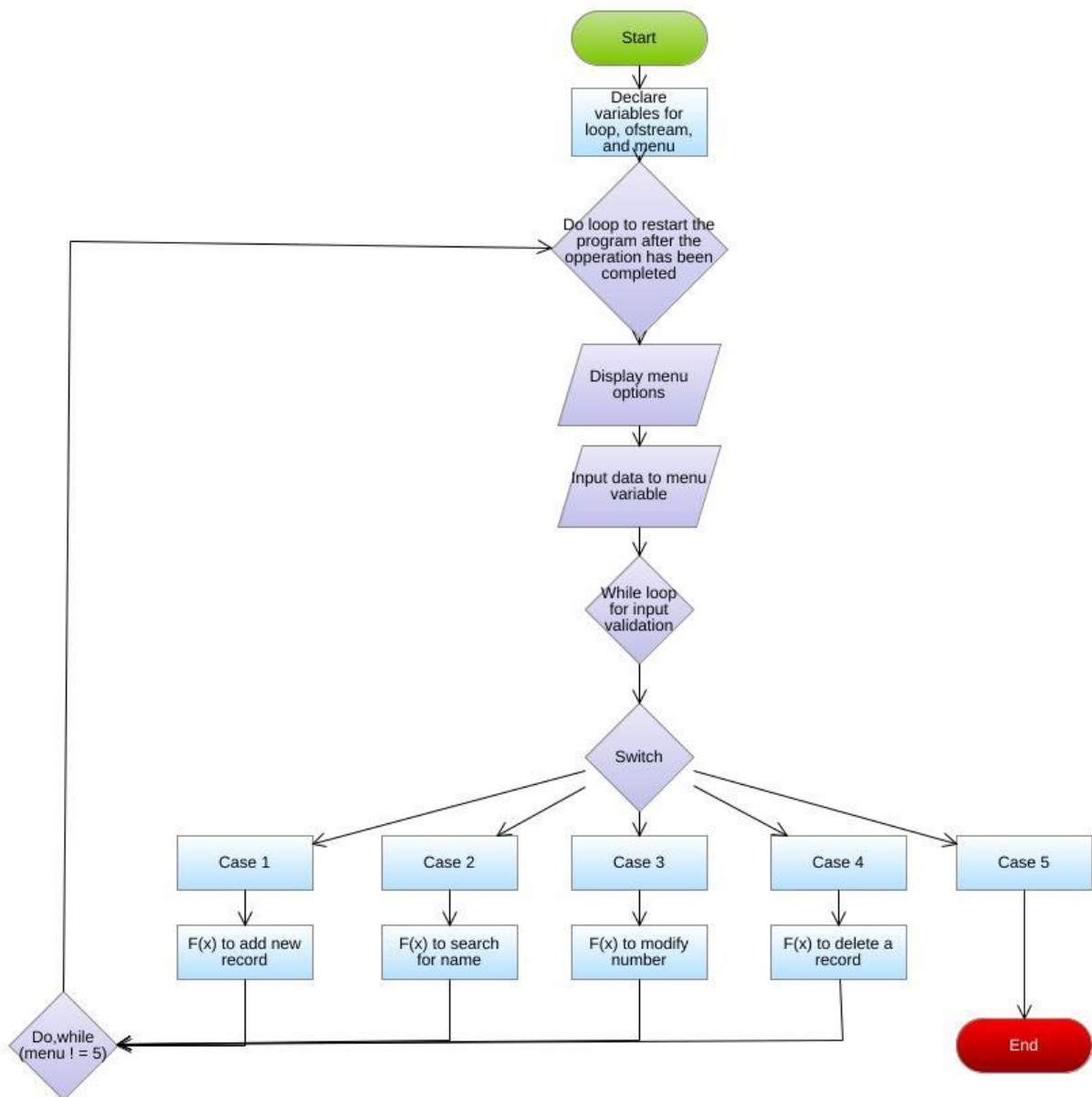
Hardware and software requirements:

The basic will run on any working computer using any C compiler.

DESIGN

Methodology: To use data structure to store phonebook data and display data.

Flowchart:

CODING IMPLEMENTATION:

```c
1    #include <stdio.h>
2    #include <stdlib.h>
3    #include <string.h>
4    #include <conio.h>
5    #include <ctype.h>
6    typedef struct list add;
7    struct list
8    {
9        char name[40];
10       char num[20];
11       char gmail[40];
12       char fb[40];
13       add *next;
14   };
15   int i = 1, size = 0;
16   void Insert(add **head)
17   {
18       add *temp = (add *)malloc(sizeof(add));
19       printf("\n\n\n\t\tContact name : ");
20       gets(temp->name);
21       system("cls");
22       printf("\n\n\n\t\tContact Name : ");
23       gets(temp->name);
24       system("cls");
25       printf("\n\n\n\t\t\tMobile Number : ");
26       gets(temp->num);
27       system("cls");
28       printf("\n\n\n\t\t\tGmail id : ");
29       gets(temp->gmail);
30       system("cls");
31       printf("\n\n\n\t\t\tFacebook Account : ");
32       gets(temp->fb);
33       system("cls");
34       temp->next = NULL;
35       size++;
36       if (size == 5)
37       {
38           system("cls");
39           printf("\n\n\n\t\t\tPhone Memory is full!!!");
```

```c
40              printf("\n\t\tIf you want to add more contact,You have to delete some contact from your list...\n\n\n");
41              return;
42          }
43          else
44          {
45              if (*head == NULL)
46              {
47                  *head = temp;
48                  system("cls");
49                  printf("\n\n\t\t\t\t\tDone!!!\n\n");
50                  return;
51              }
52              else
53              {
54                  add *p = *head;
55                  while (p->next != NULL)
56                  {
57                      p = p->next;
58                  }
59                  p->next = temp;
60                  system("cls");
61                  printf("\n\t\t\t\t\tDone!!!\n\n");
62              }
63              return;
64          }
65      }
66      void Search(add *head)
67      {
68          char ch[40];
69          printf("\n\n\n\t\tContact name : ");
70          gets(ch);
71          system("cls");
72          printf("\n\n\n\t\tContact name : ");
73          gets(ch);
74          system("cls");
75          if (head == NULL)
```

```c
76          {
77              system("cls");
78              printf("\n\n\n\t\tNo Contact exists in this Phone Book List!!!\n\n");
79
80              return;
81          }
82          else
83          {
84              add *p = head;
85              while (p != NULL)
86              {
87                  if (strcmp((p->name), ch) == 0)
88                  {
89                      system("cls");
90                      printf("\n\t\t\t%s", p->name);
91                      printf("\n\t\t-----------------------");
92                      printf("\n\t\tNumber : %s", p->num);
93                      printf("\n\t\tGmail ID : %s", p->gmail);
94                      printf("\n\t\tFacebook Account : %s\n\n", p->fb);
95                      return;
96                  }
97                  p = p->next;
98              }
99              system("cls");
100             printf("\n\n\n\t\tThis Contact is not exists in the list! ");
101         }
102 }
103
104 void Delete(add **head)
105 {
106     char ch[40];
107     printf("\n\t\tContact name : ");
108     gets(ch);
109     system("cls");
110     printf("\n\t\tContact name : ");
111     gets(ch);
112     system("cls");
```

```c
        if (*head == NULL)
        {
            system("cls");
            printf("\n\t\t\t\t\tNo Contact exists in this Phone Book List!\n\n");
            return;
        }
        else
        {
            if (strcmp(((*head)->name), ch) == 0)
            {
                add *p = *head;
                *head = (*head)->next;
                free(p);
                printf("\n\t\t\t\t\tDone!!!\n\n\n\n");
                return;
            }
            else
            {
                add *p = *head;
                while (p->next != NULL)
                {
                    if (strcmp((p->next->name), ch) == 0)
                    {
                        p->next = p->next->next;
                        size--;
                        return;
                    }
                    p = p->next;
                }
                system("cls");
                printf("\n\t\t\t\t\tInvalid Contact!!!\n\n");
            }
        }
    }
```

```c
148    void Display(add *head)
149    {
150        if (head == NULL)
151        {
152            system("cls");
153            printf("\n\n\n\t\tNo Contact exists in this Phone Book List!");
154            return;
155        }
156        else
157        {
158            add *p = head;
159            while (p != NULL)
160            {
161                printf("\n\t\t\t\t%d.%c%s", i, 32, p->name);
162                printf("\n\t\t\t-------------------------");
163                printf("\n\t\t\tNumber : %s", p->num);
164                printf("\n\t\t\tGmail ID : %s", p->gmail);
165                printf("\n\t\t\tFacebook Account : %s\n\n", p->fb);
166                p = p->next;
167                i++;
168            }
169            i = 1;
170            return;
171        }
172    }
173
```

```c
174    void main()
175    {
176        add *head = NULL;
177        char c[40];
178    mainhome:
179        system("cls");
180        printf("\n\t\t|!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!|");
181        printf("\n\t\t|                         !!!!!!!!!!!!!!!!!!!!!!!!!               |");
182        printf("\n\t\t|!!!!!!!!!!!!!!!!!!!!! WELCOME TO OUR PHONE BOOK !!!!!!!!!!!!!!!!!!|");
183        printf("\n\t\t|                     !!!!!!!!!!!!!!!!!!!!!!!!!!!!                |");
184        printf("\n\t\t|!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!|\n\n\n");
185
186        while (1)
187        {
188            printf("\n\n\t\t\t\t\tMenu");
189            printf("\n\t\t\t------------------------------------------------");
190            printf("\n\t\t\t1)Create A Contact\t\t2)Remove A Contact");
191            printf("\n\t\t\t3)Show The Contact List\t\t4)Find A Contact");
192            printf("\n\t\t\t-1)Quit");
193            printf("\n\t\t\tEnter your choice in numbers: ");
194            int ch;
195            scanf("%d", &ch);
196            if (ch == -1)
197            {
198                break;
199            }
200            else
201            {
202                switch (ch)
203                {
204                case 1:
205                    system("cls");
206                    Insert(&head);
207                    break;
208                case 2:
209                    system("cls");
210                    Delete(&head);
211                    break;
212                case 3:
```

```c
212                 case 3:
213                     system("cls");
214                     Display(head);
215                     break;
216                 case 4:
217                     system("cls");
218                     Search(head);
219                     break;
220                 default:
221                     printf("\n\t\tInvalid Choice!Try again!!!");
222                     break;
223             }
224         }
225     }
226 }
227
```

RESULT / OUTPUT

```
|!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!|
|                   !!!!!!!!!!!!!!!!!!!!!!!!!!!!!              |
|!!!!!!!!!!!!!!!!!!! WELCOME TO OUR PHONE BOOK !!!!!!!!!!!!!!!!!|
|                   !!!!!!!!!!!!!!!!!!!!!!!!!!!!!              |
|!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!|



                            Menu
              ------------------------------------------------
              1)Create A Contact        2)Remove A Contact
              3)Show The Contact List   4)Find A Contact
              -1)Quit
              Enter your choice in numbers: |
```

```
            1. abc
         --------------------------
         Number : 3448454357
         Gmail ID : sdfgdffdfsd
         Facebook Account : svfdvss



                            Menu
         ---------------------------------------------------------
         1)Create A Contact        2)Remove A Contact
         3)Show The Contact List   4)Find A Contact
         -1)Quit
         Enter your choice in numbers: |
```

```
          abc
---------------------------
Number : 3448454357
Gmail ID : sdfgdffdfsd
Facebook Account : svfdvss




                              Menu
        ----------------------------------------------------
        1)Create A Contact              2)Remove A Contact
        3)Show The Contact List         4)Find A Contact
        -1)Quit
        Enter your choice in numbers: |
```

```
                        Done!!!




                          Menu
        ----------------------------------------------------
        1)Create A Contact              2)Remove A Contact
        3)Show The Contact List         4)Find A Contact
        -1)Quit
        Enter your choice in numbers: |
```

REFERENCES:

https://www.geeksforgeeks.org/insertion-in-linked-list/

https://www.geeksforgeeks.org/search-an-element-in-a-linked-list-iterative-and-recursive/