Project topic name

Phone Book Management System

A PROJECT REPORT

Submitted by

……..

SWARNAJIT SAHOO

(210301120134)

……..

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE ENGINEERING

CENTURION UNIVERSITY OF TECHNOLOGY & MANAGEMENT

BHUBANESWAR– 752050

NOVEMBER, 2022

SPECIMEN CERTIFICATE

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING

CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT, BHUBANESWAR-752050

BONAFIDE CERTIFICATE

Certified that this project report **"…Phone-Book management system…"** is the bonafide work of **"Swarnajit Sahoo"** who carried out the project work under my supervision. This is to further certify that this project has not been carried out earlier in this institute and the university to the best of my knowledge.

(Asst. Professor of CSE department) *Certified that the above-mentioned project has been duly carried out as per the norm of the college and statutes of the university*

SIGNATURE

DEPARTMENTAL SEAL

ACKNOWLEDGEMENTS

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my/our thinking, behaviour and acts during the course of study.

We are/are thankful to Mr. Abhijit Pal for his support, cooperation, and motivation provided to me/us for constant inspiration, presence, and blessings.

We also extend our sincere appreciation to Dr Mamata Gadnayak, Head of Department, who provided valuable suggestions and precious time to accomplish our academic project.

Name: Swarnajit sahoo

Reg.No: 210301120134

OBJECTIVE

The phonebook is a very simple mini-project that can help you understand the basic concepts of functions, file handling, and data structure. This program will teach you how to add, list, change or edit, search and remove data from/to a file. Adding new records, listing them, editing and updating them, looking for saved contacts, and removing phonebook records are simple Functions that make up the main menu of this Phonebook program.

Aim of the project:

Create a "Contact Phonebook" framework using C programming. This software is very useful nowadays to store full information under a single contact number. The software also has options for removing and displaying the contact number entered.

Code

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
struct node{
    char name[40];
    char phone_no[12];
    struct node*left;   //left
    struct node*right;   //right
};
struct node*head=NULL;
char temp_n[40];
char temp_p[12];

struct node* createNode(char *name,char *phone_no){
    struct node*new_node=(struct node*)malloc(sizeof(struct node));
    strcpy(new_node->name,name);
    strcpy(new_node->phone_no,phone_no);
    new_node->left=new_node->right=NULL;
    return new_node;
}


void insert(char *name,char *phone_no){
    struct node*root=head;
    struct node *prev = NULL;
    if(root==NULL){
        struct node*new_node=createNode(name,phone_no);
        head=new_node;
        return;
    }
    while(root!=NULL){
        prev = root;
        if(strcmp(root->name,name)==0){  //string comparision
            printf("Name is already present !");
            return;
        }
        else if(strcmp(root->name,name)>0){
            root = root->left;
        }
        else{
            root = root->right;
        }
    }
    struct node* new = createNode(name,phone_no);
    if(strcmp(prev->name,name)>0){
        prev->left = new;
    }
    else{
        prev->right = new;
```

```
    }
}

struct node* inorderpre(struct node* node)
{
    while (node->right != NULL)
        node = node->right;

    return node;
}

struct node* deleteNode(struct node* root, char*name)
{
    if (root == NULL)
        return root;

    if (strcmp(root->name,name)>0)
        root->left = deleteNode(root->left, name);

    else if (strcmp(root->name,name)<0)
        root->right = deleteNode(root->right, name);

    else {
        if (root->left == NULL) {
            struct node* temp = root->right;
            free(root);
            return temp;
        }
        else if (root->right == NULL) {
            struct node* temp = root->left;
            free(root);
            return temp;
        }

        struct node* temp = inorderpre(root->left);
        strcpy(root->name,temp->name);
        root->left = deleteNode(root->left, temp->name);
    }
    return root;
}

void search(struct node*root,char *phone){
    if(root!=NULL){
        search(root->left,phone);
        if(strstr(root->phone_no,phone)){
            printf(" Name : %-12s , Phone No.: %-10s \n",root->name,root->phone_no);
        }
        search(root->right,phone);
    }
}
```

```c
void show_ascending(struct node*root){
    if(root!=NULL){
        show_ascending(root->left);
        printf(" Name : %-12s , Phone No.: %-10s \n",root->name,root->phone_no);
        show_ascending(root->right);
    }
}

void show_descending(struct node*root){
    if(root!=NULL){
        show_descending(root->right);
        printf(" Name : %-12s , Phone No.: %-10s \n",root->name,root->phone_no);
        show_descending(root->left);
    }
}

int main(){
    menu:
    printf("\n-----MENU-----\n");
    printf("1. Add Contact\n");
    printf("2. Delete Contact\n");
    printf("3. Search Contact\n");
    printf("4. Show in Ascending Order\n");
    printf("5. Show in Descending Order\n");
    printf("6. Exit Program\n");
    int choice;
    printf("Enter Choice :\n");
    scanf("%d",&choice);
    printf("\n");
    if(choice==1){
        printf("Enter Name : ");
        while((getchar())!='\n');
        scanf("%[^\n]%*c",temp_n);

        printf("Enter Phone no. : ");
        scanf("%[^\n]%*c",temp_p);
        insert(temp_n,temp_p);
    }
    else if(choice==2){
        if(head==NULL){
            printf("No contacts to delete !\n");
            goto menu;
        }
        printf("Enter Exact Name to delete : ");
        while((getchar())!='\n');
        scanf("%[^\n]%*c",temp_n);
        head=deleteNode(head,temp_n);
    }
    else if(choice==3){
```

```
        struct node*root=head;
        if(root==NULL){
            printf("\nNo Contacts to search !\n");
            goto menu;
        }
        printf("Enter Phone no. : ");
        while((getchar())!='\n');
        scanf("%[^\n]%*c",temp_p);
        printf("\n");
        search(root,temp_p);
        printf("\n");
    }
    else if(choice==4){
        struct node*root=head;
        if(root==NULL){
            printf("\nNo Contacts are there !\n");
            goto menu;
        }
        printf("\nPrinting in Ascending Order :\n");
        show_ascending(root);
        printf("\n");
    }
    else if(choice==5){
        struct node*root=head;
        if(root==NULL){
            printf("\nNo Contacts are there !\n");
            goto menu;
        }
        printf("\nPrinting in Descending Order :\n");
        show_descending(root);
        printf("\n");
    }
    else if(choice==6){
        exit(0);
    }
    else{
        printf("Enter correct number !\n");
    }
    goto menu;
return 0;
}
```

Output

Displaying contacts

```
-----MENU-----
1. Add Contact
2. Delete Contact
3. Search Contact
4. Show in Ascending Order
5. Show in Descending Order
6. Exit Program
Enter Choice :
4


Printing in Ascending Order :
 Name : raghav       , Phone No.: 45729367292
 Name : swarnajit sahoo , Phone No.: 9120594120
```

Deleting a contact

```
-----MENU-----
1. Add Contact
2. Delete Contact
3. Search Contact
4. Show in Ascending Order
5. Show in Descending Order
6. Exit Program
Enter Choice :
2

Enter Exact Name to delete : raghav

-----MENU-----
1. Add Contact
2. Delete Contact
3. Search Contact
4. Show in Ascending Order
5. Show in Descending Order
6. Exit Program
Enter Choice :
4


Printing in Ascending Order :
 Name : swarnajit sahoo , Phone No.: 9120594120
```

CONCLUSION

The application program has been successfully implemented using experimental cases and the language used – C. This application works for other functions that make it easy to search, delete, edit, and remember our peer information.

The main purpose of this package is to reduce the pressure on users to learn more about computers and software. This helps maintain a direct connection between the computer and the user. This “My Phonebook” feature allows users with other logs to add and easily search for buttons and search options.