

Add Contact

Exit

List Contact

Block

Unblock

Delete contact

Modify Contact

Search contact

**Introduction of Digital Phonebook**

1. **Project Abstract:** Phonebook system is a small web application developed for mini project students. Digital phonebook is a new system, by using this application we can store all the details in a central repository. By using this application we cans see our contacts anywhere in the world. In this project we cab save our contacts and we can search them by name and we can see all of them at a time.
2. **Feature:** In this digital phonebook we can add new contacts, view all the contact list, modify the contact list, search contact, delete any contact and then we can exit the program.
3. **Working:** We use digital phonebook for talking with each other by saving the number in the contact list.
4. **Future update:**  In future, we will add recording system, image save system etc.
5. **Limitation:**
6. In adding names, we cannot take two words without space.
7. We can take only g-mail address.
8. **Bonus:**
9. List names in alphabetical order
10. Background Display color
11. Searching for single character

**Source Code:**

**#include<stdio.h>**

**#include<stdlib.h>**

**#include<string.h>**

**void menu();**

**void add\_contacts();**

**void show\_list();**

**void Delete();**

**void modify();**

**void search();**

**void block();**

**void block\_list();**

**void unblock();**

**struct PhoneBook**

**{**

**char name[100];**

**char address[200];**

**char phoneNo[200];**

**char email[100];**

**};**

**const char\* Email()**

**{**

**char email[30];**

**char \*p;**

**p = email;**

**scanf("%s",email);**

**int i=0;**

**while(email[i] != '\0')**

**{**

**i++;**

**}**

**i = i-10;**

**if(email[i] != '@' || email[i+1] != 'g' || email[i+2] != 'm' || email[i+3] != 'a' || email[i+4] != 'i' || email[i+5] != 'l'**

**|| email[i+6] != '.' || email[i+7] != 'c' || email[i+8] != 'o' || email[i+9] != 'm')**

**{**

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

**printf("\n\t\t\t\tPlease Enter Your Valid Email ");**

**Email();**

**}**

**else**

**return p;**

**}**

**void add\_contacts()**

**{**

**system("cls");**

**system("COLOR 8E");**

**FILE \*fp;**

**fp=fopen("phonebook.txt","a");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**else**

**{**

**struct PhoneBook ph;**

**char name[100];**

**char address[200];**

**char phoneNo[200];**

**char email[100];**

**printf("\n\t\t\t\t\* Here you can add new contact \* ");**

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

**printf("\n\n\t\t\t\tEnter New Name : ");**

**scanf("%s", ph.name);**

**printf("\n\t\t\t\tEnter Address : ");**

**scanf("%s", ph.address);**

**printf("\n\t\t\t\tEnter E-Mail : ");**

**(strcpy(ph.email,Email()));**

**printf("\n\t\t\t\tEnter phoneNo : ");**

**scanf("%s", ph.phoneNo);**

**if(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tPhone No must be 11 digits");**

**while(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tEnter phoneNo : ");**

**scanf("%s", ph.phoneNo);**

**if(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tPhone No must be 11 digits");**

**}**

**}**

**}**

**fprintf(fp, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**fclose(fp);**

**char choice;**

**printf("\n\t\t\t\t\* Add Another Contact -> press Y/y\n\t\t\t\t\* For main menu -> press any key.\n");**

**choice=getche();**

**while((choice=='Y')||(choice=='y'))**

**{**

**add\_contacts();**

**}**

**if(choice!='Y'||(choice!='y'))**

**{**

**system("cls");**

**menu();**

**}**

**}**

**void show\_list()**

**{**

**system("cls");**

**system("COLOR 60");**

**FILE \*fp;**

**fp=fopen("phonebook.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**else**

**{**

**struct PhoneBook phonebook;**

**int size=0;**

**while(fscanf(fp,"%s %s %s %s", phonebook.name, phonebook.address, phonebook.email, phonebook.phoneNo)!=EOF)**

**{**

**size++;**

**}**

**rewind(fp);**

**struct PhoneBook pb[size];**

**int i=0;**

**while(fscanf(fp,"%s %s %s %s", phonebook.name, phonebook.address, phonebook.email, phonebook.phoneNo)!=EOF)**

**{**

**strcpy(pb[i].name,phonebook.name);**

**strcpy(pb[i].address,phonebook.address);**

**strcpy(pb[i].email,phonebook.email);**

**strcpy(pb[i].phoneNo,phonebook.phoneNo);**

**i++;**

**}**

**for(int i=0; i<size; i++)**

**{**

**for(int j=0; j<size; j++)**

**{**

**if(strcmp(strupr(pb[i].name), strupr(pb[j].name))<0)**

**{**

**struct PhoneBook temp;**

**strcpy(temp.name,pb[i].name);**

**strcpy(temp.address,pb[i].address);**

**strcpy(temp.email,pb[i].email);**

**strcpy(temp.phoneNo,pb[i].phoneNo);**

**strcpy(pb[i].name,pb[j].name);**

**strcpy(pb[i].address,pb[j].address);**

**strcpy(pb[i].email,pb[j].email);**

**strcpy(pb[i].phoneNo,pb[j].phoneNo);**

**strcpy(pb[j].name,temp.name);**

**strcpy(pb[j].address,temp.address);**

**strcpy(pb[j].email,temp.email);**

**strcpy(pb[j].phoneNo,temp.phoneNo);**

**}**

**}**

**}**

**{**

**printf("\n\t\t\t\t\* Here is the list you saved \* ");**

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

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

**}**

**for(int i=0; i<size; i++)**

**{**

**printf("\n\t\t\t\tName : %s \n\t\t\t\tAddress : %s \n\t\t\t\tE-Mail: %s\n\t\t\t\tPhone No: %s \n\n",pb[i].name,pb[i].address, pb[i].email, pb[i].phoneNo);**

**}**

**}**

**fclose(fp);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void Delete()**

**{**

**system("cls");**

**system("COLOR 4F");**

**struct PhoneBook ph;**

**FILE \*fp;**

**FILE \*temp;**

**int count=0;**

**char name[50];**

**fp=fopen("phonebook.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**temp=fopen("TEMP.txt","w");**

**printf("\n\t\t\t\* Here you can delete any contact \* ");**

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

**printf("\n\n\t\tEnter Contact Name you want to Delete : ");**

**scanf("%s",&name);**

**rewind(fp);**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**if( strcmp(strupr(ph.name),strupr(name))==0)**

**{**

**count++;**

**continue;**

**}**

**else**

**{**

**fprintf(temp, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**}**

**fclose(fp);**

**fclose(temp);**

**remove ("phonebook.txt") ;**

**rename ("TEMP.txt", "phonebook.txt") ;**

**if(count!=0)**

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

**else**

**printf("\n\t\t\t\tNo entry found with Name : %s",name);**

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

**fclose(fp);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void modify()**

**{**

**system("cls");**

**system("COLOR F8");**

**struct PhoneBook ph;**

**FILE \*fp;**

**FILE \*temp;**

**int count=0;**

**char name[50];**

**fp=fopen("phonebook.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**temp=fopen("TEMP.txt","w");**

**printf("\n\t\t\t\t\* Here you can modify any contact \*");**

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

**printf("\n\n\t\t\tEnter Contant Name you want to modify : ");**

**scanf("%s",&name);**

**rewind(fp);**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**if( strcmp(strupr(ph.name),strupr(name))==0)**

**{**

**printf("\t\t\tYou are updating : ");**

**printf("\n\n\t\t\t\tEnter Name : ");**

**scanf("%s", ph.name);**

**printf("\n\t\t\t\tEnter Address : ");**

**scanf("%s", ph.address);**

**printf("\n\t\t\t\tEnter E-Mail : ");**

**(strcpy(ph.email,Email()));**

**printf("\n\t\t\t\tEnter phoneNo : ");**

**scanf("%s", ph.phoneNo);**

**if(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tPhone No must be 11 digits");**

**while(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tEnter phoneNo : ");**

**scanf("%s", ph.phoneNo);**

**if(strlen(ph.phoneNo) != 11)**

**{**

**printf("\n\t\t\t\tPhone No must be 11 digits");**

**}**

**}**

**}**

**count++;**

**fprintf(temp, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**else**

**{**

**fprintf(temp, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**}**

**fclose(fp);**

**fclose(temp);**

**remove ("phonebook.txt") ;**

**rename ("TEMP.txt", "phonebook.txt") ;**

**if(count!=0)**

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

**else**

**printf("\n\t\t\t\tNo entry found with Name : %s",name);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void search()**

**{**

**system ("cls");**

**system("COLOR 30");**

**struct PhoneBook ph;**

**int result;**

**int count=0;**

**char target[50];**

**FILE \*fp;**

**fp=fopen("phonebook.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**printf("\n\t\t\t\* Here you can search any contact \*");**

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

**printf("\n\n\t\tEnter the name you want to search : ");**

**scanf("%s",&target);**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**result=checkForMatch(strupr(ph.name),strupr(target));**

**if(result==1)**

**{**

**count++;**

**printf("\n\t\t\t\tName : %s \n\t\t\t\tAddress : %s \n\t\t\t\tE-Mail: %s\n\t\t\t\tPhone No: %s \n\n",ph.name,ph.address, ph.email, ph.phoneNo);**

**}**

**}**

**fclose(fp);**

**if(count==0)**

**printf("\n\t\tNothing found for Name : %s",target);**

**else**

**printf("\n\t\tTotal %d no of Contact(s) found for : %s",count,target);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**int checkForMatch(char a[],char b[])**

**{**

**int i,k;**

**char \*p,temp[strlen(b)];**

**if(strlen(a)>=strlen(b))**

**{**

**for(i=0; i<=(strlen(a)-strlen(b)); i++)**

**{**

**p=&a[i];**

**k=0;**

**while(k!=strlen(b))**

**{**

**temp[k]=\*(p+k);**

**k++;**

**}**

**temp[k]='\0';**

**if(strcmp(strupr(temp),strupr(b))==0)**

**{**

**return 1;**

**}**

**}**

**}**

**return 0;**

**}**

**void block()**

**{**

**system("cls");**

**system("color F4");**

**struct PhoneBook ph;**

**printf("\t\t\t\t\* Here you can block your contact \*");**

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

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

**char target[50];**

**int count=0;**

**printf("\t\t\t Enter name : ");**

**scanf("%s",target);**

**FILE \*fp,\*fp1,\*fp2;**

**fp=fopen("phonebook.txt","r");**

**fp1=fopen("block.txt","a");**

**fp2=fopen("temp.txt","w");**

**if(fp==NULL || fp1==NULL || fp2==NULL)**

**{**

**printf("Error opening file");**

**}**

**rewind(fp);**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**if( strcmp(strupr(ph.name),strupr(target))==0)**

**{**

**fprintf(fp1, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**count++;**

**}**

**else**

**{**

**fprintf(fp2, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**}**

**fclose(fp);**

**fclose(fp1);**

**fclose(fp2);**

**remove ("phonebook.txt") ;**

**rename ("TEMP.txt", "phonebook.txt") ;**

**if(count!=0)**

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

**else**

**printf("\n\t\t\t\tNo entry found with Name : %s",target);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void block\_list()**

**{**

**system("cls");**

**system("color 4F");**

**struct PhoneBook ph;**

**printf("\t\t\t\t\* Here you can see block numbers : \*");**

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

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

**FILE \*fp;**

**fp=fopen("block.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**printf("\n\t\t\t\t# Blocked Contacts : ");**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**printf("\n\t\t\t\tName : %s \n\t\t\t\tAddress : %s \n\t\t\t\tE-Mail: %s\n\t\t\t\tPhone No: %s \n\n",ph.name,ph.address, ph.email, ph.phoneNo);**

**}**

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

**rewind(fp);**

**fclose(fp);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void unblock()**

**{**

**system("cls");**

**system("color 4F");**

**struct PhoneBook ph;**

**printf("\t\t\t\t\* Here you can unblock your Contact \*");**

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

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

**char target[50];**

**int count=0;**

**FILE \*fp,\*fp1,\*fp2;**

**fp=fopen("block.txt","r");**

**if(fp==NULL)**

**{**

**printf("Error opening file");**

**}**

**printf("\n\t\t\t\t# Blocked Contacts : ");**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**printf("\n\t\t\t\tName : %s \n\t\t\t\tAddress : %s \n\t\t\t\tE-Mail: %s\n\t\t\t\tPhone No: %s \n\n",ph.name,ph.address, ph.email, ph.phoneNo);**

**}**

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

**rewind(fp);**

**fclose(fp);**

**printf("\t\t\t Enter name You Want to Unblock : ");**

**scanf("%s",target);**

**fp=fopen("block.txt","r");**

**fp1=fopen("phonebook.txt","a");**

**fp2=fopen("temp.txt","w");**

**while(fscanf(fp,"%s %s %s %s", ph.name, ph.address, ph.email, ph.phoneNo)!=EOF)**

**{**

**if( strcmp(strupr(ph.name),strupr(target))==0)**

**{**

**fprintf(fp1, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**count++;**

**}**

**else**

**{**

**fprintf(fp2, "%s %s %s %s\n", ph.name, ph.address, ph.email, ph.phoneNo);**

**}**

**}**

**fclose(fp);**

**fclose(fp1);**

**fclose(fp2);**

**remove ("block.txt") ;**

**rename ("TEMP.txt", "block.txt") ;**

**if(count!=0)**

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

**else**

**printf("\n\t\t\t\tNo entry found with Name : %s",target);**

**char choice;**

**printf("\n\t\t\t\t\* For main menu -> press 1\n\t\t\t\t\* For exit -> press any key.\n");**

**printf("\n\t\t\t\tYour Choice : ");**

**choice=getche();**

**if(choice == '1')**

**{**

**menu();**

**}**

**else**

**{**

**exit(0);**

**}**

**}**

**void menu()**

**{**

**system("cls");**

**system("COLOR E0");**

**char choice;**

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

**printf("\n\t\t\t\t\t EAST WEST UNIVERSITY");**

**printf("\n\t\t\t\t\t Department of CSE");**

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

**printf("\n\t\t\t\t\tWELCOME TO DIGITAL PHONEBOOK");**

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

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

**printf("\t\t\t# What do you want to do ? :)");**

**printf("\n\n\t\t\t\t[1] ADD NEW CONTACT");**

**printf("\n\t\t\t\t[2] VIEW ALL CONTACTS");**

**printf("\n\t\t\t\t[3] SEARCH CONTACTS");**

**printf("\n\t\t\t\t[4] MODIFY CONTACTS");**

**printf("\n\t\t\t\t[5] DELETE CONTACT");**

**printf("\n\t\t\t\t[6] BLOCK");**

**printf("\n\t\t\t\t[7] BLOCK LIST");**

**printf("\n\t\t\t\t[8] UNBLOCK");**

**printf("\n\t\t\t\t[9] EXIT");**

**printf("\n\n\n\t\t\t ENTER YOUR CHOISE --> ");**

**choice=getche();**

**switch(choice)**

**{**

**case '1' :**

**add\_contacts();**

**break;**

**case '2':**

**show\_list();**

**break;**

**case '3':**

**search();**

**break;**

**case '4':**

**modify();**

**break;**

**case '5':**

**Delete();**

**break;**

**case '6':**

**block();**

**break;**

**case '7':**

**block\_list();**

**break;**

**case '8':**

**unblock();**

**break;**

**case '9':**

**printf("\n\n\t\t\t\t\tTHANK YOU FOR USING OUR PHONEBOOK\n");**

**exit(0);**

**break;**

**default:**

**system("cls");**

**printf("ERROR !! PRESS 1-9 ONLY");**

**}**

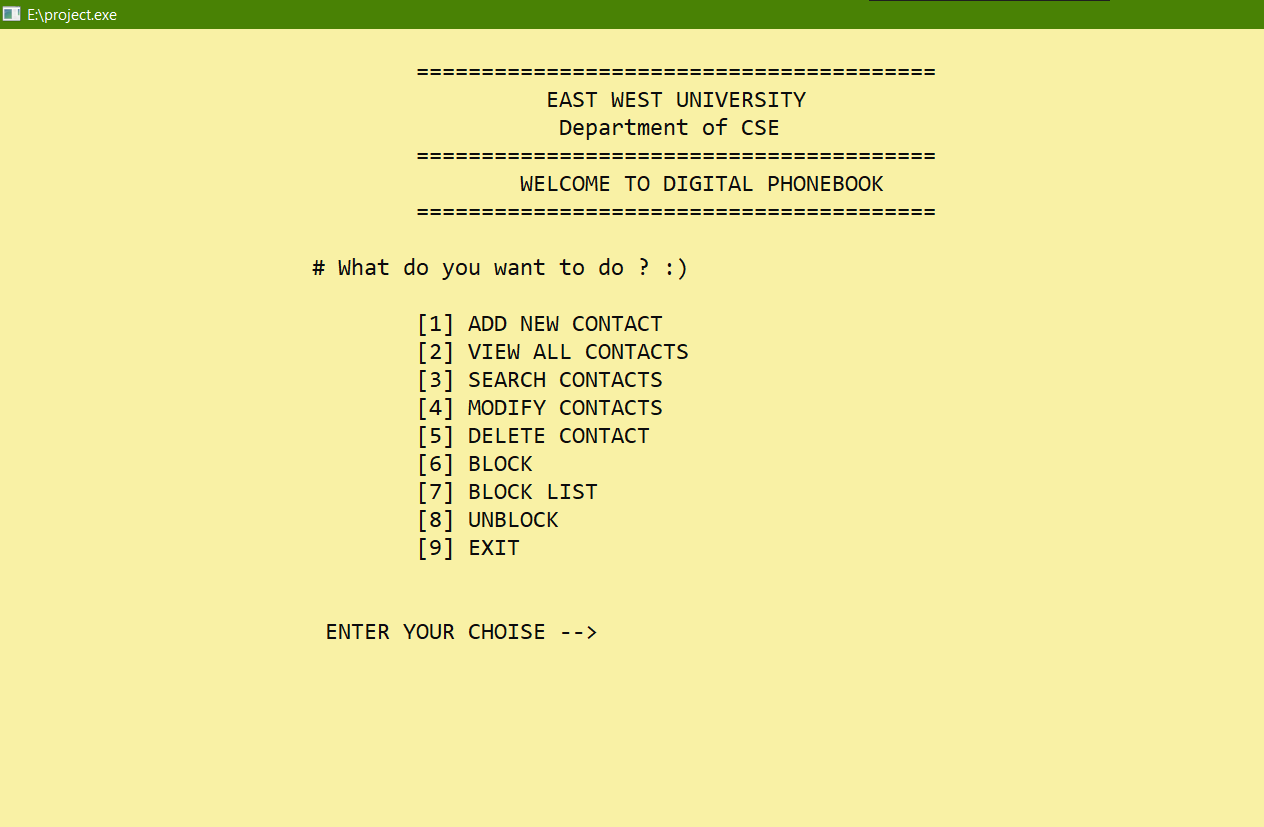
**}**

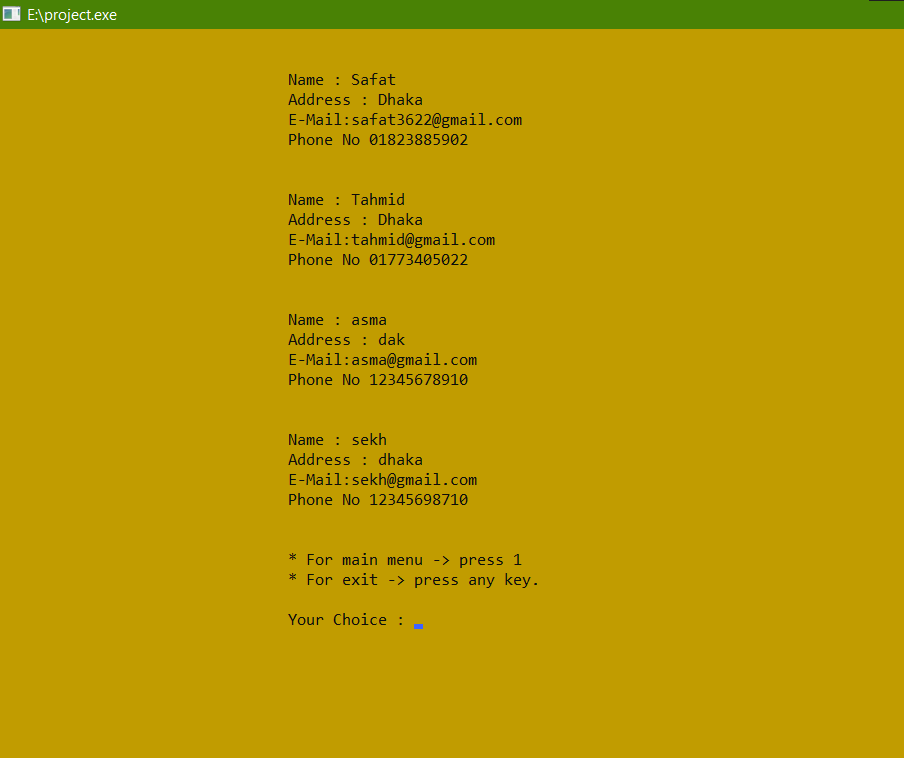
**int main()**

**{**

**menu();**

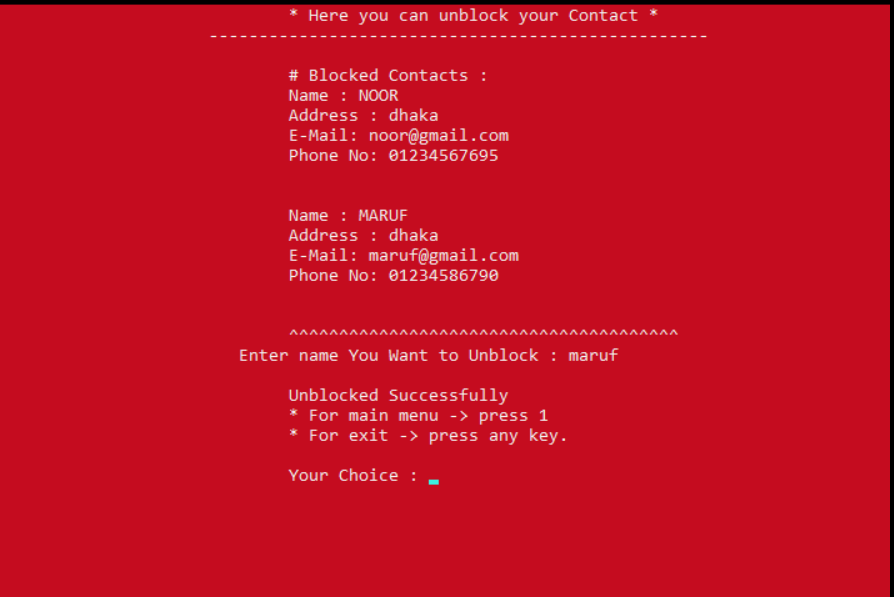
**}**

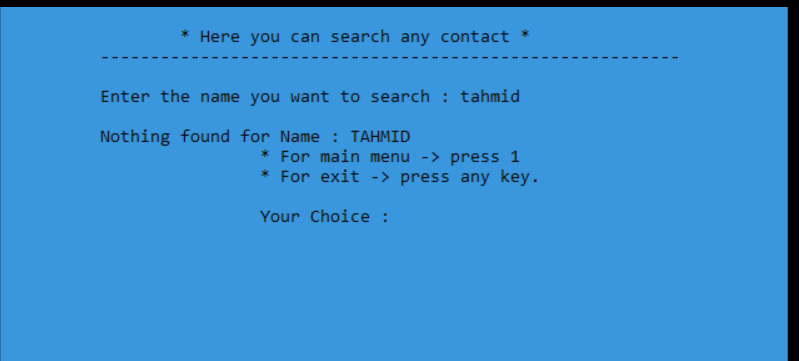


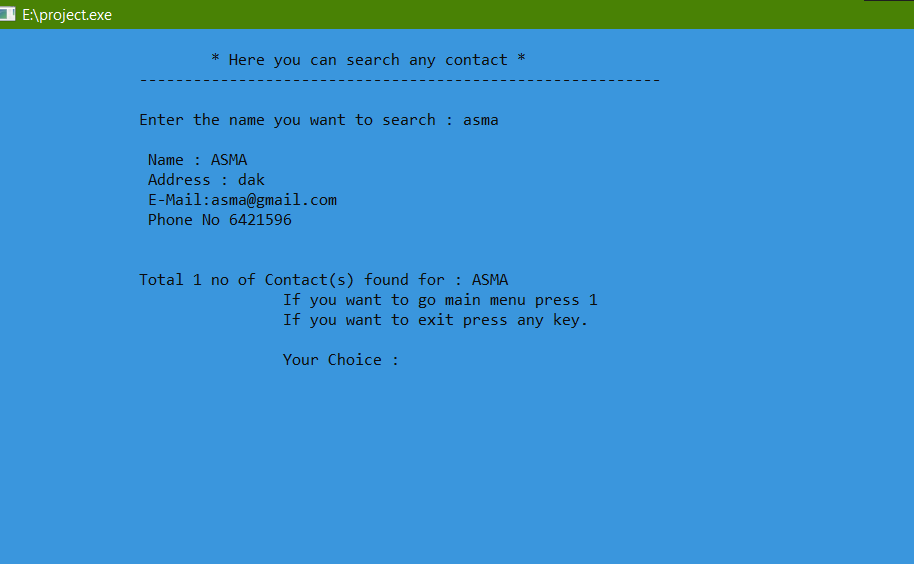


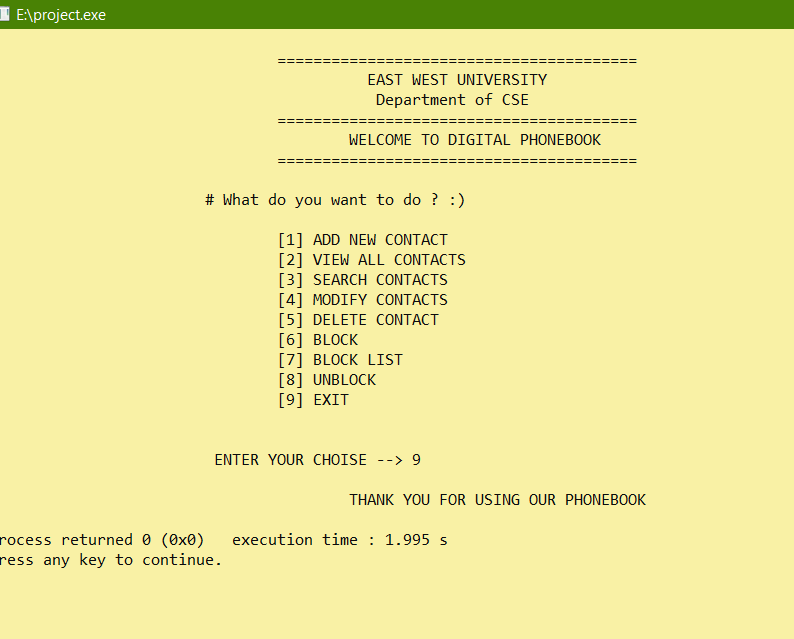


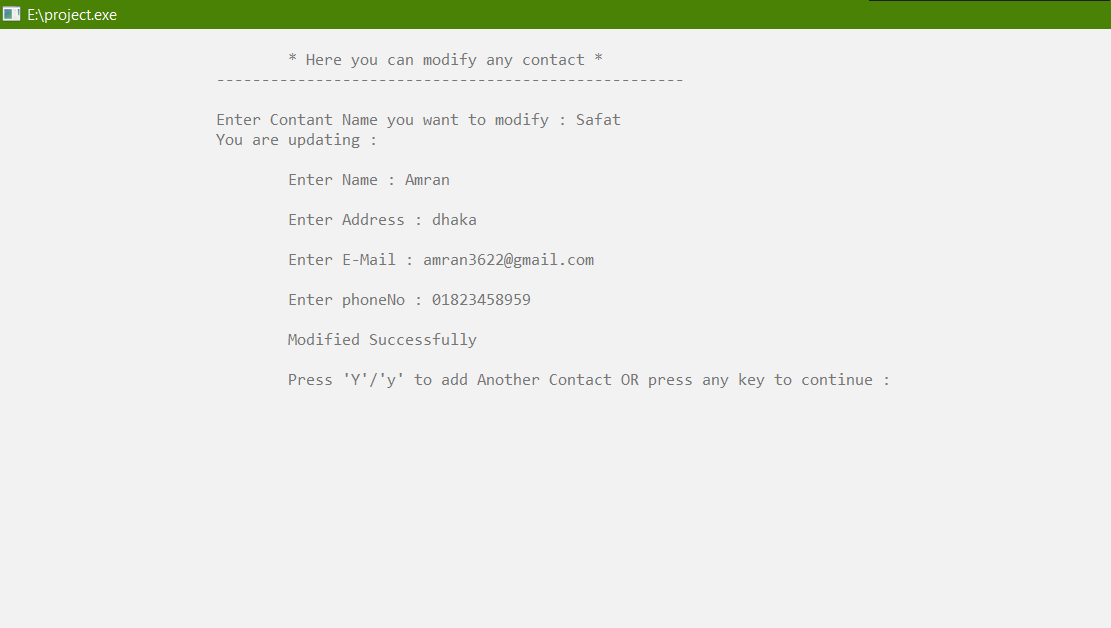


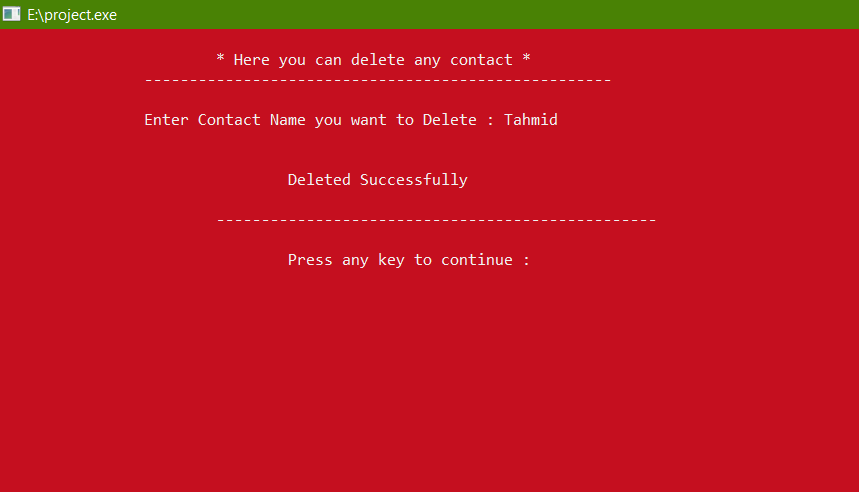












**CSE103 – Structured Programming**

**Project Evaluation Rubrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Max.** |  | **Awarded** |
|  | |  |  |  |
| **A. Report** | |  |  |  |
|  |  |  |  |  |
| i. | Introduction / Problem statement |  |  |  |
|  |  |  |  |  |
| iv. | System Design |  |  |  |
|  |  |  |  |  |
| v. | Program output (Screen shots) |  |  |  |
|  |  |  |  |  |
| vi. | Source code |  |  |  |
|  | |  |  |  |
| vii. Disk/CD neatly attached (Y/N) | |  |  |  |
|  | |  |  |  |
|  | |  |  |  |
| **B. Source Code** | |  |  |  |
|  |  |  |  |  |
| i. | Style |  |  |  |
|  |  |  |  |  |
|  | Indentation |  |  |  |
|  |  |  |  |  |
|  | Self-documentation |  |  |  |
|  |  |  |  |  |
| ii. | Modularity (small size functions) |  |  |  |
|  |  |  |  |  |
| iii. | Error reporting capabilities |  |  |  |
|  |  |  |  |  |
| iv. | Code efficiency, strategy, and originality |  |  |  |
|  | |  |  |  |
|  | |  |  |  |
| **C. Program Execution** | |  |  |  |
|  |  |  |  |  |
| i. | Compile without errors |  |  |  |
|  |  |  |  |  |
| ii. | User friendly |  |  |  |
|  |  |  |  |  |
| iii. | Error free during runtime |  |  |  |
|  |  |  |  |  |
| iv. | Program output |  |  |  |
|  | |  |  |  |
|  | |  |  |  |
| **D. Presentation and Demonstration [Psychomotor Domain]** | |  |  |  |
|  | |  |  |  |
| i.  Presentation and communication skills (**Soft skill)** | |  |  |  |
|  | |  |  |  |
|  | |  |  |  |
| **E. Bonus** | |  |  |  |
|  |  |  |  |  |
| i. | Extra significant features |  |  |  |
|  |  |  |  |  |
|  | **TOTAL** |  | **11** |  |
|  |  |  |  |  |

**CSE103 – Structured Programming**

**Project Declaration**

**(Student 1)**

**Student ID:** 2021-3-60-022

**Name:** Amran Hossain

**Session:** 2021-Fall

**Project No.:** 04. Digital Phonebook

**Date submitted:** 10.01.22

**Deadline of the project:** 11.01.22

**My contribution in doing this**

**project (in percentage) in the group:** 33.3%

**Description of my contribution in this**

**project in the group:** Gave reasonable advice. I assist write the code. Contact Adding & Searching Contact.

**Number of hours I spent in doing this**

**Project:** 10 Days.

**(Student 2)**

**Student ID:** 2021-3-60-024

**Name:** Rafsun Islam Taskin

**Session:** 2021-Fall

**Project No.:** 04. Digital Phonebook

**Date submitted:** 10.01.22

**Deadline of the project:** 11.01.22

**My contribution in doing this**

**project (in percentage) in the group:** 33.3%

**Description of my contribution in this**

**project in the group:** Code writing , Modify & Delete Contact., block unblock.

**Number of hours I spent in doing this**

**Project:** 10 Days.

**(Student 3)**

**Student ID:** 2021-3-60-026

**Name:** Mohammad Tahmid Noor

**Session:** 2021-Fall

**Project No.:** 04. Digital Phonebook

**Date submitted:** 10.01.22

**Deadline of the project:** 11.01.22

**My contribution in doing this**

**project (in percentage) in the group:** 33.3%

**Description of my contribution in this**

**project in the group:** Listing contact in Alphabetical Order & Code editing Designing interface.

**Number of hours I spent in doing this**

**Project:** 10 Days.

**We hereby certify that this project represents the work done by all our group members with our contribution clearly stated above without copying from any other resources. We declare that no part of our work has been copied from or by other groups, and that no collusion has taken place with any other persons or groups.**

**We certify that any disks submitted with this project have been virus checked and have no viruses on them.**

(1) Signature: …………………………… Date: ........................................

(2) Signature: …………………………… Date: ........................................

(3) Signature: …………………………… Date: ........................................

Thank You

* Team even