

**PROBLEM SOLVING WITH PROGRAMMING [RA20]**

(PSP)

**B.TECH 1st YEAR 2nd SEMESTER**

**PROJECT TITLE :- PHONEBOOK**

**TEAM NO. 11**

**COURSE PROJECT**

(2021-22)

Developed By:

**Chakravarthi [HTNO:2105A41022]**

**Nikhil Reddy [HTNO:2103A51064]**

**Hrushith [HTNO:2103A51084]**

**Harshini [HTNO:2103A51053]**

**Section: A2**

Under the Guidance of

# Dr.P.Praveen

# SR University

Warangal.

|  |  |
| --- | --- |
| contents | Pg-no |
| 1.Module description  2.Abstract  3.Implementation  4.output&conclusion | 3-5  6  7-18  19 |

# MODULE DESCRIPTION:-

# 1.Stdio.h :

# printf : It is used to print the strings, integer, character etc on the output screen.

# scanf : It reads the character, string, integer etc from the keyboard.

# getc : It reads the character from the file.

# putc : It writes the character to the file.

# fopen : It opens the file and all file handling functions are defined in stdio.h header file.

# fclose : It closes the opened file.

# remove : It deletes the file.

# fflush : It flushes the file.

# 2.Stdlib.h :

# Communication with the Environment functions

# about : Abort program

# atexit : Register function to be called at program exit

# getenv : Get Environment String

# system : Perform Operating System Command

# Integer Arithmetic functions

# abs : Absolute Value of Integer

# div : Integer Division

# labs : Absolute Value of Long Integer

# ldiv : Long Integer Division

# Dynamically Allocated Array functions

# calloc : Allocate and Clear Memory Block

# malloc : Allocate Memory Block

# realloc : Resize Memory Block

# 

# 3.String.h :

# strcat : Concentrates str2 at the end of str1

# strncat : Appends a portion of string to another

# strcpy : Copies str2 into str1

# strncpy : Copies given number of characters of one string to another

# strlen : Gives the length of str1

# strcmp : Returns 0 if str1 is same as str2. Returns <0 if strl<str2. Return >0 if str1>str2

# strcmpi : Same as strcmp() function. But, this fumction negotiates case. “A” and “a” are treated as same.

# strchr : Returns pointer to first occurrence of char in str1

# strrchr : last occurrence of given character in a string is found

# strstr : Returns pointer to first occurrence of str2 in str1

# strrstr : Returns pointer to last occurrence of str2 in str1

# strdup : Duplicates the string

# strlwr : Converts string to lowercase

# strupr : Converts string to uppercase

# strrev : Reverse the given string

# strnset : It sets the portion of characters in a string to given character

# strtok : Tokenizing given string using delimiter

# ABOUT:-

# 

# By using this file, we can contract and do these operators in it.

* + View all contacts.
  + Add a contact.
  + Remove details of any contact number.
  + Search details of any contact.
  + Update details of any contact.
  + Delete whole phonebook (delete all contacts present in phonebook.)

***In this project, you will find a very good implementation of file handling using C programming.***

***Below are the topics which are implemented in this project.***

-Serialization.

-Storing and retriving the block of data into a file.

-Secure data by converting it into binary format.

-Pointers.

# PHONEBOOK

**CODE:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct person

{

char name[30];

char country\_code[4];

long int mble\_no;

char sex[8];

char hallticketno[100];

};

typedef struct person person;

void remove\_all();

void print\_menu();

void add\_person();

void list\_record();

void search\_person();

void remove\_person();

void update\_person();

void start();

void take\_input(person \*p);

int main()

{

start();

return 0;

}

void start()

{

int choice;

while(1)

{

print\_menu();

scanf("%d",&choice);

switch(choice)

{

case 1:

list\_record();

getchar();

getchar();

break;

case 2:

add\_person();

getchar();

getchar();

break;

case 3:

search\_person();

getchar();

getchar();

break;

case 4:

remove\_person();

getchar();

getchar();

break;

case 5:

update\_person();

getchar();

getchar();

break;

case 6:

remove\_all();

getchar();

getchar();

break;

default :

system("clear");

printf("Thanks for using phonebook visit again )\n");

getchar();

getchar();

exit(1);

}

}

}

void print\_menu()

{

system("clear");

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

printf("\t\t\*Welcome TO My phone book\*\n");

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

printf("\t\t\t1) list record\n\n");

printf("\t\t\t2) Add person\n\n")

printf("\t\t\t3) Search person Details\n\n");

printf("\t\t\t4) Remove person\n\n");

printf("\t\t\t5) Update person\n\n");

printf("\t\t\t6) Delete all contacts\n\n");

printf("\t\t\t7) exit Phonebook\n\n\n");

printf("\t\t\t\tEnter your Choice : ");

}

void add\_person()

{

system("clear");

FILE \*fp;

fp = fopen("phonebook\_db", "ab+");

if (fp == NULL)

{

printf("Error in file opening, Plz try again !\n");

printf("Press any key to continue....\n");

return;

}

else

{

person p;

take\_input(&p);

fwrite(&p, sizeof(p), 1, fp);

fflush(stdin);

fclose(fp);

system("clear");

printf("Record added Successfully\n");

printf("Press any key to continue ....\n");

}

{

void take\_input(person \*p)

{

system("clear");

getchar();

printf("Enter name : ");

scanf("%[^\n]s",p->name);

printf("Enter country code : ");

scanf("%s",p->country\_code);

printf("Enter mobile no : ");

scanf("%ld",&p->mble\_no);

printf("Enter sex : ");

scanf("%s",p->sex);

printf("Enter email : ");

scanf("%s",p->hallticketno);

}

void list\_record()

{

system("clear");

FILE \*fp;

fp = fopen("phonebook\_db", "rb");

if (fp == NULL)

{

printf("Error in file opening, Plz try again !\n");

printf("Press any key to continue....\n");

return;

}

else

{

person p;

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

printf("\t\t\t\t\* Here is all records listed in phonebook \*\n");

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

printf(" NAME\t\t\t\t COUNTRY CODE\t\t PHONE NO\t\t SEX\t\t HALL TICKET NO\n");

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

// fseek(fp,-(sizeof(p)\*2L),2);

while (fread(&p, sizeof(p), 1, fp) == 1)

{

int i;

int len1 = 40 - strlen(p.name);

int len2 = 19 - strlen(p.country\_code);

int len3 = 15;

int len4 = 21 - strlen(p.sex);

printf("%s",p.name);

for(i=0;i<len1;i++) printf(" ");

printf("%s",p.country\_code);

for(i=0;i<len2;i++) printf(" ");

printf("%ld",p.mble\_no);

for(i=0;i<len3;i++) printf(" ");

printf("%s",p.sex);

for(i=0;i<len4;i++) printf(" ");

printf("%s",p.hallticketno);

printf("\n");

fflush(stdin);

}

fflush(stdin);

fclose(fp);

printf("\n\nPress any key to continue....\n");

}

}

void search\_person()

{

system("clear");

long int phone;

printf("Enter Phone number of the person you want to search : ");

scanf("%ld",&phone);

FILE \*fp;

fp = fopen("phonebook\_db", "rb");

if (fp == NULL)

{

printf("Error in file opening, Plz try again !\n");

printf("Press any key to continue....\n");

return;

}

else

{

int flag = 0;

person p;

while (fread(&p, sizeof(p), 1, fp) == 1)

{

if(p.mble\_no == phone)

{

printf(" NAME\t\t\t\t COUNTRY CODE\t\t PHONE NO\t\t SEX\t\t HALL TICKET NO\n");

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

int i;

int len1 = 40 - strlen(p.name);

int len2 = 19 - strlen(p.country\_code);

int len3 = 15;

int len4 = 21 - strlen(p.sex);

printf("%s",p.name);

for(i=0;i<len1;i++) printf(" ");

printf("%s",p.country\_code);

for(i=0;i<len2;i++) printf(" ");

printf("%ld",p.mble\_no);

for(i=0;i<len3;i++) printf(" ");

printf("%s",p.sex);

for(i=0;i<len4;i++) printf(" ");

printf("%s",p.hallticketno);

printf("\n");

flag = 1;

break;

}

else continue;

// fflush(stdin);

}

if(flag == 0)

{

system("clear");

printf("Person is not in the Phonebook\n");

}

fflush(stdin);

fclose(fp);

printf("\n\nPress any key to continue....\n");

}

}

void remove\_person()

{

system("clear");

long int phone;

printf("Enter Phone number of the person you want to remove from phonebook : ");

scanf("%ld",&phone);

FILE \*fp,\*temp;

fp = fopen("phonebook\_db", "rb");

temp = fopen("temp","wb+");

if (fp == NULL)

{

printf("Error in file opening, Plz try again !\n");

printf("Press any key to continue....\n");

return;

}

else

{

person p;

int flag = 0;

while (fread(&p, sizeof(p), 1, fp) == 1)

{

if(p.mble\_no == phone)

{

system("clear");

printf("Person removed successfully\n");

flag = 1;

}

else fwrite(&p,sizeof(p),1,temp);

fflush(stdin);

}

if(flag == 0)

{

system("clear");

printf("No record found for %d number\n",phone);

}

fclose(fp);

fclose(temp);

remove("phonebook\_db");

rename("temp","phonebook\_db");

fflush(stdin);

printf("Press any key to continue....\n");

}

}

void update\_person()

{

system("clear");

long int phone;

printf("Enter Phone number of the person you want to update details : ");

scanf("%ld",&phone);

FILE \*fp,\*temp;

fp = fopen("phonebook\_db", "rb");

temp = fopen("temp","wb+");

if (fp == NULL)

{

printf("Error in file opening, Plz try again !\n");

printf("Press any key to continue....\n");

return;

}

else

{

int flag = 0;

person p;

while (fread(&p, sizeof(p), 1, fp) == 1)

{

if(p.mble\_no == phone)

{

take\_input(&p);

fwrite(&p, sizeof(p), 1, temp);

system(“clear”);

printf("Data updated successfully\n");

flag = 1;

}

else fwrite(&p,sizeof(p),1,temp);

fflush(stdin);

}

if(flag == 0)

{

system("clear");

printf("No record found for %d number\n",phone);

}

fclose(fp);

fclose(temp);

remove("phonebook\_db");

rename("temp","phonebook\_db");

fflush(stdin);

printf("Press any key to continue....\n");

}

}

void remove\_all()

{

char choice;

system("clear");

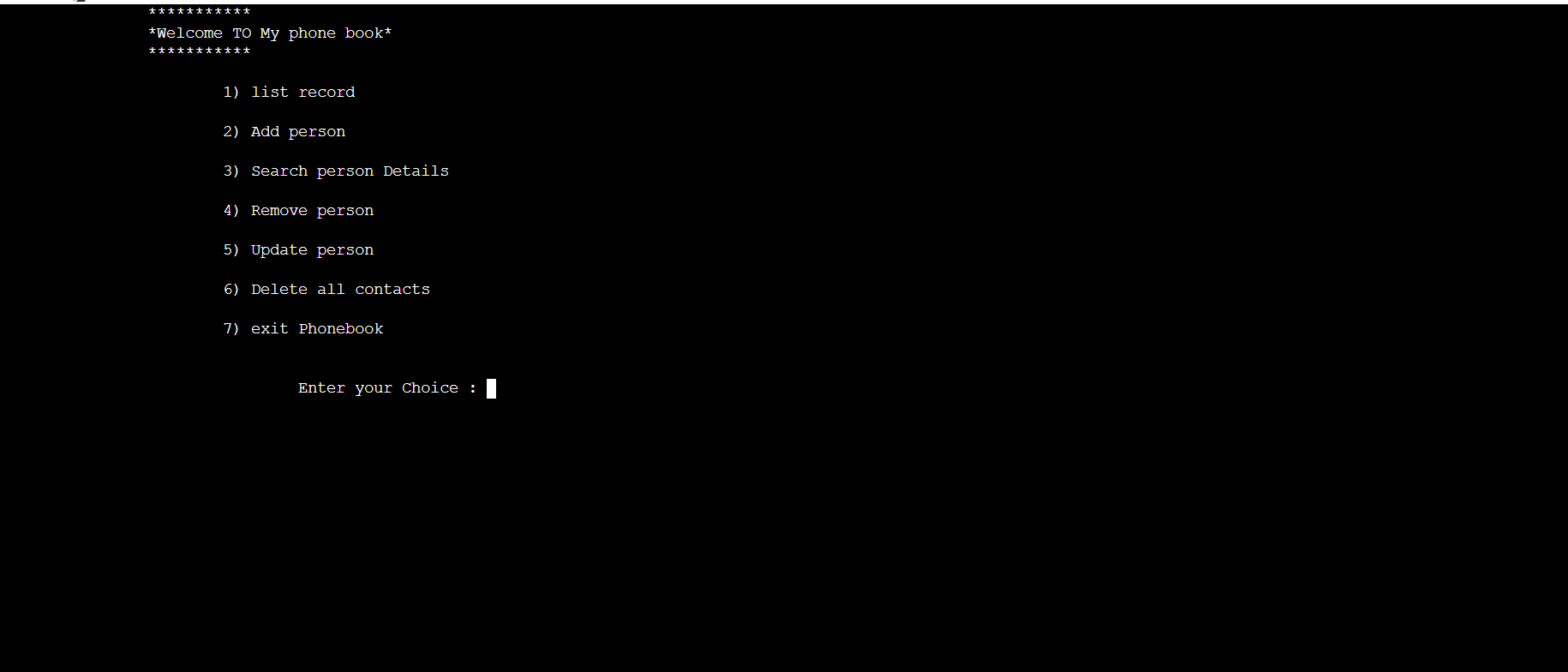
remove("./phonebook\_db");

printf("All data in the phonebook deleted successfully\n");

printf("Press any key to continue ... \n");

}

**OUTPUT:**

****

# CONCLUSION:-

***In this project, you will find a very good implementation of file handling using C programming.***

**THANKYOU**