

# [Buddy book essay](https://assignbuster.com/buddy-book-essay/)

Contents

* 10. Appendix

### Buddy book

### Problem Solving & A ; Program Design Using C

### 2. Introduction

The aim of this assignment is to make an application to keep a list of friends contact inside informations utilizing c. plan is chiefly meant for users to identify in their names and contact inside informations, like a phone book. User can add, edit, delete and hunt contacts utilizing portion of their information. Users should be able to quite the bill of fare and add more records whenever they wish to.

To be successful in this undertaking, we have to do a Buddy book which is user friendly and error free as possible.

### 3. Undertaking description

Our chief aim is to do a brother book which allow users to add contacts, redact them, hunt for a contact by a missive or name, display all contacts and take a contact from the list..

Main bill of fare of the plan provide 5 options.

1. Add a friend

2. Modify friends ‘ inside informations

3. Display all my friends

4. Search friends

5. Exit

### Adding a contact

When adding a new contact to buddy book user is promote to come in these inside informations.

Enter First Name:

Enter Last Name:

Enter Home Number:

Enter nomadic Number:

Make you wish to add more Numberss to phonebook ( Y/N ) ?

If the user choice is “ Y” above process is reiterating. If user choice is “ N” user redirected to chief bill of fare.

### Modifying a contact

When a user selects this option it will expose all the saved contacts in buddy book and prompt user to choose a contact by contact order. After choosing the contact figure to redact, plan will give these options to choose.

1. Full Contact Press 1

2. First Name Press 2

3. Last Name Press 3

4. Home Number Press 4

5. Mobile Number imperativeness 5

6. Travel to chief bill of fare imperativeness 6

7. Exit imperativeness 7

Selection 1-This gives user to modify a full contact. He can modify all the Fieldss in the contact.

Choice 2-if he selects figure two he can merely modify first name of the contact.

Selection 3- If he selects figure three he can merely modify Last name of the contact.

Selection 4- If he selects figure four he can merely modify Home figure of the contact.

Selection 5- If he selects figure five he can merely modify nomadic figure of the contact.

Selection 6- If he selects figure six he will airt to chief bill of fare.

Selection 7- If he selects figure seven plan will go out.

### Display all contacts

This will expose the full contacts shop in buddy book. If there is n’t any contacts store in buddy book, it will expose

Phonebook is empty ; make you desire to add contacts? ( Y/N )

If user selects ‘ Y ‘ he will motivate to add contact bill of fare. If he select ‘ N ‘ he will motivate to chief bill of fare.

### Search contacts

In hunt bill of fare there are two options user can choose to make a hunt

1. Search by first missive

2. Search by first name

1. Search by first missive

Users prompt to come in a character and the plan will expose all records with names that start with that character.

2. Search by first name

This will expose all records starts with that entered name

### 5. Pseudo codification

Get down

read pick

Case pick

instance 1: Add friends

instance 2: Modify friends inside informations

instance 3: Display all my friends

instance 4: Search friends

instance 5: Exit

default: show “ Invalid choice ”

End instance

read count= 0

If ( choice== 1 )

If ( count== 10 )

show “ You have exceeded the brother book capacity ”

Else

show “ Enter the first name ”

read firstName

show “ Enter the last name ”

read lastName

show “ Enter the place figure ”

read homeNum

show “ Enter the nomadic figure ”

read mobileNum

count= count+1

show “ Data successfully saved! ! ! ”

show “ Do you wish to add more Numberss ( Y/N ) ? ”

read respond

If ( respond==’Y ‘ )

return to instance 1

Else

return to chief bill of fare

End if

End if

Else if ( choice== 2 )

show “ Phone book contact list ”

i= 0

while ( one & lt ; = count )

show firstName

show lastName

show homeNum

show mobileNum

i++

End while

show “ Enter the figure of the contact that you want to modify ”

read respond

If ( respond & gt ; count ) or ( respond & lt ; 0 )

show “ You have entered an invalid figure. Check and enter the right figure ”

Else

show “ If you want to modify: ”

show “ Full contact Press1 ”

show “ First name Press2 ”

show “ Last name Press3 ”

show “ Home figure Press4 ”

show “ Mobile figure Press5 ”

show “ Enter your pick ”

read respond

If ( respond== 1 )

show “ contact you want to modify is: ”

show firstName

show lastName

show homeNum

show mobileNum

show “ Now come in your new informations ”

show “ Enter new first name ”

read firstName

show “ Enter new last name ”

read lastName

show “ Enter new place figure ”

read homeNum

show “ Enter new nomadic figure ”

read mobileNum

show “ Data successfully saved! ! ! ”

show “ Do you desire to modify an another contact ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

return to instance 2

Else

return to chief bill of fare

End if

End if

Else if ( respond== 2 )

show “ contact you want to modify is: ”

show firstName

show lastName

show homeNum

show mobileNum

show “ Now come in your new informations ”

show “ Enter new first name ”

read firstName

show “ Data successfully saved! ! ! ”

show “ Do you desire to modify an another contact ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

return to instance 2

Else

return to chief bill of fare

End if

Else if ( respond== 3 )

show “ contact you want to modify is: ”

show firstName

show lastName

show homeNum

show mobileNum

show “ Now come in your new informations ”

show “ Enter new last name ”

read lastName

show “ Data successfully saved! ! ! ”

show “ Do you desire to modify an another contact ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

return to instance 2

Else

return to chief bill of fare

End if

Else if ( respond== 4 )

show “ contact you want to modify is: ”

show firstName

show lastName

show homeNum

show mobileNum

show “ Now come in your new informations ”

show “ Enter new place figure ”

read homeNum

show “ Data successfully saved! ! ! ”

show “ Do you desire to modify an another contact ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

return to instance 2

Else

return to chief bill of fare

End if

Else if ( respond== 5 )

show “ contact you want to modify is: ”

show firstName

show lastName

show homeNum

show mobileNum

show “ Now come in your new informations ”

show “ Enter new nomadic figure ”

read mobileNum

show “ Data successfully saved! ! ! ”

show “ Do you desire to modify an another contact ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

return to instance 2

Else

return to chief bill of fare

End if

Else

show “ Invalid codification ”

End if

Else if ( choice== 3 )

show “ Phone book contact list ”

i= 0

while ( one & lt ; = count )

show firstName

show lastName

show homeNum

show mobileNum

i++

End while

Else if ( choice== 4 )

read first\_char

read given name

show “ Search ”

show “ If you want to seek by: ”

show “ first missive Press1 ”

show “ first name Press2 ”

show “ Enter your choice ”

read choice

If ( choice & lt ; 0 ) or ( choice & gt ; 2 )

show “ Invalid codification ”

Else if ( selection== 1 )

show “ hunt by first missive of the name ”

show “ Enter foremost missive to hunt ”

read first\_letter

i= 0

while ( one & lt ; = count )

i++

If ( first\_letter== first\_char )

show firstName

show lastName

show homeNum

show mobileNum

Else

show “ No lucifers found ”

End if

End while

Else

show “ hunt by first name ”

show “ Enter foremost name to hunt ”

read given name

i= 0

while ( one & lt ; = count )

i++

If ( first\_name== firstName )

show firstName

show lastName

show homeNum

show mobileNum

Else

show “ No lucifers found ”

End if

End while

End if

Else if ( choice== 5 )

show “ Are you sure that you want to go out ( Y/N ) ? ”

read reply

If ( answer==’Y ‘ )

abort

Else

return to chief bill of fare

End if

Else

show “ Invalid choice ”

show “ Check the bill of fare and come in your pick once more ”

End if

End

### 7. Limitations

Here we are traveling to state some things about the restrictions in the plan that we developed.

In this plan we had to restrict the figure of contacts to 6. An effort to add more contacts would give an mistake message.

We applied some restrictions when come ining informations. In the instance of figure of characters besides we had to restrict the sum. For the place figure and office figure Fieldss merely whole numbers can be applied and limit that to 15 whole numbers in each field.

Here we can seek a contact merely by first missive and first name. we ca n’t seek a contact by last name or a contact figure.

In this phone book we ca n’t add a in-between name of a individual to the phone book.

In our brother book merely add two Numberss for a individual. If individual have more than two Numberss there are n’t any manner to add all Numberss into a one contact.

### 8. FURTHER Development

There are some restrictions in the plan that we have developed. One restriction is the sum of contacts that we included in the plan. In a hereafter development we hope to increase the sum of contacts.

In the plan that we developed there are a few Fieldss such as first name, Last name, place figure, nomadic figure etc. In hereafter we would wish to increase these by including Fieldss like e-mail reference and place reference of a contact.

Further we hope to present more methods of seeking contacts in a hereafter development. There are restrictions in the computing machine linguistic communication we used. If we are to utilize a computing machine linguistic communication with graphical user interface, we would be able to present much more synergistic computing machine plan. Therefore in future we would wish to utilize a linguistic communication with GUI.

If we use programming linguistic communications which support GUI, we can set an option to add a existent exposure of a user to his contact inside informations.

Besides we can add a information economy option to the buddy book. So users can hive away contacts and can utilize it in future.

We can add a information sorting option to this plan. So when a user hunt something consequences automatically sorted and show to user. So it becomes more user-friendly.

### 9. Decision

It is a great pleasance for us to holding got an chance to show some of our thoughts at successful decision of this undertaking. This undertaking gave us an chance to set into pattern some of the lessons that we were taught theoretically at the talk room. We obtained a huge cognition when we engaged practically in this undertaking. We put into pattern the units Functions, Flow charts, Arrays, Structures, Pseudo codes. These are really of import lessons for us In C Language.

This undertaking gave us an chance to work as a squad to accomplish a common end. With that we learnt how to work as a squad with a squad sprit to carry through a common end.

### 10. Appendix

# include & lt ; stdio. h & gt ; // Standard C input & A ; end product library

# include & lt ; conio. h & gt ; //use getch bid

# include & lt ; stdlib. h & gt ; // abort programme by utilizing abort ( ) bid

# include & lt ; string. h & gt ; // String maps

# define T 1

# define F 0

# define MAX 6

//prototype of contact inside informations

struct buddybook // Declaration of buddybook construction

{

char firstname [ 50 ] ;

char lastname [ 50 ] ;

char homenum [ 15 ] ;

char mobilenum [ 15 ] ;

} ;

struct buddybook bbook [ MAX ] ;

int entry= 0 ;

//declareing pototype of maps

nothingness bill of fare ( ) ;

nothingness attention deficit disorder ( ) ;

nothingness edit ( ) ;

nothingness show ( ) ;

null hunt ( ) ;

null graphic ( ) ;

nothingness chief ( )

{

in writing ( ) ;

printf ( “ press any key to travel to the chief bill of fare of the phone book ” ) ;

getch ( ) ;

for ( ; ; )

{

bill of fare ( ) ;

}

}

null graphic ( ) ////declareing f maps

{

printf ( “ ttkkkkkkkkkkkkkkkkkkkktt” ) ;

printf ( “ kkkkkkkkkkkkttttttttkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkk kkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkktt kkkkkkkkkkkk” ) ;

printf ( “ ttkkkkkkkkkkkk kkkkkkkkkkkktt” ) ;

printf ( “ kkkkkkkkkkkkkk ttttkkkkkkkkkkkkkk” ) ;

printf ( “ ttkkkkkkkkkkkkkk ttkkkkkkkkkkkkkkkkkkkktt” ) ;

printf ( “ ttkkkkkkkkkkkkkk kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkkkktt kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkkkktt kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkkkktt kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkkkktt kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ ttkkkkkkkkkkkkkk kkkkkkkkkkkkkkkkkkkkkkkk” ) ;

printf ( “ tkkkkkkkkkkkkkk kkkkkkkkkkkkkkkkkkkkkktt” ) ;

printf ( “ kkkkkkkkkkkkkk ttttttttkkkkkkkkkkkkkk” ) ;

printf ( “ ttkkkkkkkkkkkk tttttttkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkktt kkkkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkk kkkkkkkkkk” ) ;

printf ( “ kkkkkkkkkkkktttt ttkkkkkkkk” ) ;

printf ( “ ttkkkkkkkkkkkkkkkkkkkktt” ) ;

printf ( “” ) ;

printf ( “” ) ;

printf ( “ BUDDY BOOK” ) ;

printf ( “ © by Aloka, chathura, Isuri

” ) ;

}

nothingness bill of fare ( )

{

clrscr ( ) ;

int pick ;

printf ( “ \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*” ) ;

printf ( “ \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*” ) ;

printf ( “ \*\* \*\*” ) ;

printf ( “ \*\* MAIN MENU \*\*” ) ;

printf ( “ \*\* \*\*” ) ;

printf ( “ \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*” ) ;

printf ( “ \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

” ) ;

printf ( “ 1. Add Contacts” ) ;

printf ( “ 2. Modify Contacts” ) ;

printf ( “ 3. Display all Records” ) ;

printf ( “ 4. Search” ) ;

printf ( “ 5. Exit” ) ;

printf ( “ what do you desire to make? : “ ) ;

scanf ( “ % vitamin D ” , & A ; pick ) ;

fflush ( stdin ) ;

switch ( pick ) //switching statements

{

instance 1: // User wants to add an entry

attention deficit disorder ( ) ;

interruption ;

instance 2: // User wants to redact an entry

edit ( ) ;

interruption ;

instance 3: // User wants to expose entry

show ( ) ;

interruption ;

instance 4: // User wants to seek contacts

hunt ( ) ;

interruption ;

instance 5: // User wants to go out from plan

abort ( ) ;

interruption ;

default:

clrscr ( ) ; //clears the text temper window

printf ( “ Invalid choice” ) ;

printf ( “ Press any cardinal to return to the Main Menu ” ) ;

getch ( ) ; //get a character from the keyboard

bill of fare ( ) ;

interruption ;

}

}

nothingness attention deficit disorder ( )

{

clrscr ( ) ;

if ( entry== MAX )

{

printf ( “ \*\* You have exeeded the phone book capacity \*\*” ) ;

printf ( “

Press any Key to goto Main Menu ” ) ;

getch ( ) ;

clrscr ( ) ;

return ;

}

else

{

char choice ;

clrscr ( ) ;

printf ( “ ADD NEW CONTACT” ) ;

printf ( “ Enter the First Name: ” ) ;

gets ( bbook [ entry ] . firstname ) ;

fflush ( stdin ) ;

printf ( “ Enter the Last Name: ” ) ;

gets ( bbook [ entry ] . lastname ) ;

fflush ( stdin ) ;

printf ( “ Enter the Home Number: ” ) ;

gets ( bbook [ entry ] . homenum ) ;

fflush ( stdin ) ;

printf ( “ Enter the Mobile Number: ” ) ;

gets ( bbook [ entry ] . mobilenum ) ;

fflush ( stdin ) ;

entry++ ;

clrscr ( ) ;

printf ( “ Data successfully saved! !” ) ;

printf ( “ Do you wish to add more Numberss to phonebook ( Y/N ) ? “ ) ;

scanf ( “ % degree Celsius ” , & A ; choice ) ;

fflush ( stdin ) ;

if ( ( selection==’y ‘ ) || ( selection==’Y ‘ ) )

{

attention deficit disorder ( ) ;

}

else if ( ( selection==’n ‘ ) || ( selection==’N ‘ ) )

{

return ; //exit from the current map

}

}

clrscr ( ) ;

}

nothingness edit ( )

{

clrscr ( ) ;

int ten ;

int selection2 ;

int choice1 ;

char choice2 ;

char selection3 ;

char issue ;

printf ( “ \*\* CONTACT LIST \*\*” ) ;

for ( x= 0 ; x & lt ; entry ; x++ )

{

printf ( “[ % vitamin D ] . Name: % s % s” , x+1, bbook [ x ] . firstname, bbook [ x ] . lastname ) ;

printf ( “ place Number: % s” , bbook [ x ] . homenum ) ;

printf ( “ nomadic figure: % s” , bbook [ x ] . mobilenum ) ;

}

printf ( “ Enter the Contact Number You Want to Modified: ” ) ;

printf ( “

Enter a Number between 01- % vitamin D: “ , entry ) ;

scanf ( “ % vitamin D ” , & A ; selection2 ) ;

if ( selection2 & lt ; 1 )

{

clrscr ( ) ;

printf ( “ Invalid choice” ) ;

printf ( “

press any cardinal to goto chief bill of fare ” ) ;

getch ( ) ;

return ;

}

if ( selection2 & gt ; entry )

{

clrscr ( ) ;

printf ( “ Invalid choice” ) ;

printf ( “

press any cardinal to goto chief bill of fare ” ) ;

getch ( ) ;

bill of fare ( ) ;

}

else

{

clrscr ( ) ;

printf ( “ \*\* EDIT contacts \*\*” ) ;

printf ( “

If you want to modified the: ” ) ;

printf ( “ Full Contact Press 1 ” ) ;

printf ( “ First Name Press 2 ” ) ;

printf ( “ Last Name Press 3 ” ) ;

printf ( “ Home Number Press 4 ” ) ;

printf ( “ nomadic Number imperativeness 5 ” ) ;

printf ( “ Return to Main bill of fare imperativeness 6 ” ) ;

printf ( “ EXIT imperativeness 7 ” ) ;

printf ( “

what do you desire to make? “ ) ;

scanf ( “ % vitamin D ” , & A ; choice1 ) ;

switch ( choice1 )

{

`case 1:

clrscr ( ) ;

printf ( “ \*\* MODIFIED THE FULL CONTACT \*\*” ) ;

printf ( “ Contact you traveling to modified is: ” ) ;

printf ( “[ % vitamin D ] . Name: % s % s” , selection2, bbook [ selection2-1 ] . firstname, bbook [ selection2-1 ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ selection2-1 ] . homenum ) ;

printf ( “ Mobile Number: % s” , bbook [ selection2-1 ] . mobilenum ) ;

printf ( “

Do you desire to Continue? ?” ) ;

printf ( “ If Yes imperativeness ‘ Y ‘ “ ) ;

printf ( “ If No imperativeness any Key “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; selection3 ) ;

fflush ( stdin ) ;

if ( ( selection3==’y ‘ ) || ( selection3==’Y ‘ ) )

{

clrscr ( ) ;

printf ( “

\*. Enter New First Name ” ) ;

gets ( bbook [ selection2-1 ] . firstname ) ;

fflush ( stdin ) ;

printf ( “\*. Enter New Last Name ” ) ;

gets ( bbook [ selection2-1 ] . lastname ) ;

fflush ( stdin ) ;

printf ( “\*. Enter New Home Number ” ) ;

gets ( bbook [ selection2-1 ] . homenum ) ;

fflush ( stdin ) ;

printf ( “\*. Enter New nomadic figure ” ) ;

gets ( bbook [ selection2-1 ] . mobilenum ) ;

fflush ( stdin ) ;

clrscr ( ) ;

printf ( “ Data successfully saved! ! ! \*\*” ) ;

printf ( “ Do you desire to modify an another contact “ ) ;

printf ( “ If yes Press Y “ ) ;

printf ( “ If No imperativeness any other cardinal “ ) ;

printf ( “ Enter your Choice: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; choice2 ) ;

fflush ( stdin ) ;

if ( ( choice2==’y ‘ ) || ( choice2==’Y ‘ ) )

{

edit ( ) ;

}

else

{

return ;

}

}

else

{

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

interruption ;

instance 2:

clrscr ( ) ;

printf ( “ MODIFIED THE FIRST NAME OF THE CONTACT” ) ;

printf ( “ Contact your traveling to modified is: ” ) ;

printf ( “[ % vitamin D ] . Name: % s % s” , selection2, bbook [ selection2-1 ] . firstname, bbook [ selection2-1 ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ selection2-1 ] . homenum ) ;

printf ( “ Mobile Number: % s” , bbook [ selection2-1 ] . mobilenum ) ;

printf ( “ Do you desire to Countinue? ? “ ) ;

printf ( “ If Yes imperativeness ‘ Y ‘ “ ) ;

printf ( “ If No imperativeness any Key “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; selection3 ) ;

fflush ( stdin ) ;

if ( ( selection3==’y ‘ ) || ( selection3==’Y ‘ ) )

{

clrscr ( ) ;

printf ( “

Enter New Contact First Name: “ ) ;

gets ( bbook [ selection2-1 ] . firstname ) ;

fflush ( stdin ) ;

printf ( “

” ) ;

clrscr ( ) ;

printf ( “ \*\* Data successfully saved! ! ! \*\*” ) ;

printf ( “ Do you desire to modify an another contact “ ) ;

printf ( “ If yes Press Y “ ) ;

printf ( “ If No imperativeness any other cardinal “ ) ;

printf ( “ Enter your Choice: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; choice2 ) ;

fflush ( stdin ) ;

if ( ( choice2==’y ‘ ) || ( choice2==’Y ‘ ) )

{

edit ( ) ;

}

else

{

return ;

}

}

else

{

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

interruption ;

instance 3:

clrscr ( ) ;

printf ( “ MODIFIED THE LAST NAME OF THE CONTACT” ) ;

printf ( “ Contact you traveling to modified is: ” ) ;

printf ( “[ % vitamin D ] . Name: % s % s” , selection2, bbook [ selection2-1 ] . firstname, bbook [ selection2-1 ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ selection2-1 ] . homenum ) ;

printf ( “ Mobile Number: % s” , bbook [ selection2-1 ] . mobilenum ) ;

printf ( “ Do you desire to Countinue? ? “ ) ;

printf ( “ If Yes imperativeness ‘ Y ‘ “ ) ;

printf ( “ If No imperativeness any Key “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; selection3 ) ;

fflush ( stdin ) ;

if ( ( selection3==’y ‘ ) || ( selection3==’Y ‘ ) )

{

clrscr ( ) ;

printf ( “ Enter New Contact Last Name ” ) ;

gets ( bbook [ selection2-1 ] . lastname ) ;

fflush ( stdin ) ; ;

clrscr ( ) ;

printf ( “ \*\* Data successfully saved! ! ! \*\*” ) ;

printf ( “ Do you desire to modify an another contact “ ) ;

printf ( “ If yes Press Y “ ) ;

printf ( “ If No imperativeness any other cardinal “ ) ;

printf ( “ Enter your Choice: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; choice2 ) ;

fflush ( stdin ) ;

if ( ( choice2==’y ‘ ) || ( choice2==’Y ‘ ) )

{

edit ( ) ;

}

else

{

return ;

}

}

else

{

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

interruption ;

instance 4:

clrscr ( ) ;

printf ( “ \*\* MODIFIED THE HOME NUMBER OF THE CONTACT \*\*” ) ;

printf ( “ Contact you traveling to modified is: ” ) ;

printf ( “[ % vitamin D ] . Name: % s % s” , selection2, bbook [ selection2-1 ] . firstname, bbook [ selection2-1 ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ selection2-1 ] . homenum ) ;

printf ( “ Mobile Number: % s” , bbook [ selection2-1 ] . mobilenum ) ;

printf ( “ Do you desire to Countinue? ? “ ) ;

printf ( “ If Yes imperativeness ‘ Y ‘ “ ) ;

printf ( “ If No imperativeness any Key “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; selection3 ) ;

fflush ( stdin ) ;

if ( ( selection3==’y ‘ ) || ( selection3==’Y ‘ ) )

{

clrscr ( ) ;

fflush ( stdin ) ;

printf ( “ Enter New Contact Home Number ” ) ;

gets ( bbook [ selection2-1 ] . homenum ) ;

fflush ( stdin ) ;

clrscr ( ) ;

printf ( “ \*\* Data successfully saved! ! ! \*\*” ) ;

printf ( “ Do you desire to modify an other contact “ ) ;

printf ( “ If yes Press Y “ ) ;

printf ( “ If No imperativeness any other cardinal “ ) ;

printf ( “ Enter your Choice: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; choice2 ) ;

fflush ( stdin ) ;

if ( ( choice2==’y ‘ ) || ( choice2==’Y ‘ ) )

{

edit ( ) ;

}

else

{

return ;

}

}

else

{

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

interruption ;

instance 5:

clrscr ( ) ;

printf ( “ \*\* MODIFIED THE MOBILE NUMBER OF THE CONTACT \*\*” ) ;

printf ( “ Contact you traveling to modified is: ” ) ;

printf ( “[ % vitamin D ] . Name: % s % s” , selection2, bbook [ selection2-1 ] . firstname, bbook [ selection2-1 ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ selection2-1 ] . homenum ) ;

printf ( “ Mobile Number: % s” , bbook [ selection2-1 ] . mobilenum ) ;

printf ( “ Do you desire to Countinue? ? “ ) ;

printf ( “ If Yes imperativeness ‘ Y ‘ “ ) ;

printf ( “ If No imperativeness any Key “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; selection3 ) ;

fflush ( stdin ) ;

if ( ( selection3==’y ‘ ) || ( selection3==’Y ‘ ) )

{

clrscr ( ) ;

printf ( “ Enter New nomadic figure ” ) ;

gets ( bbook [ selection2-1 ] . mobilenum ) ;

fflush ( stdin ) ;

clrscr ( ) ;

printf ( “ \*\* Data successfully saved! ! ! \*\*” ) ;

printf ( “

Do you desire to modify an another contact “ ) ;

printf ( “ t If yes Press Y “ ) ;

printf ( “ If No imperativeness any other cardinal “ ) ;

printf ( “ Enter your Choice: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; choice2 ) ;

fflush ( stdin ) ;

if ( ( choice2==’y ‘ ) || ( choice2==’Y ‘ ) )

{

edit ( ) ;

}

else

{

return ;

}

}

else

{

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

interruption ;

instance 6:

bill of fare ( ) ;

interruption ;

instance 7:

clrscr ( ) ;

printf ( “

Do you want to Exit from the Programme ( Y/N ) ?” ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; issue ) ;

fflush ( stdin ) ;

if ( ( exit==’y ‘ ) || ( exit==’Y ‘ ) )

{

abort ( ) ;

}

else

{

clrscr ( ) ;

printf ( “ Press any Key to goto the chief bill of fare ” ) ;

getch ( ) ;

bill of fare ( ) ;

}

default:

clrscr ( ) ;

printf ( “ \*\* You Have Entered Invalid Number \*\*” ) ;

printf ( “ Press any Key to goto the edit bill of fare ” ) ;

getch ( ) ;

edit ( ) ;

}

}

}

nothingness show ( )

{

char selection1 ;

int num ;

char user\_inp ;

if ( entry== 0 )

{

clrscr ( ) ;

printf ( “ phone book is empty” ) ;

printf ( “ Do you desire to add contacts ( Y/N ) ? “ ) ;

scanf ( “ % degree Celsius ” , & A ; selection1 ) ;

if ( ( selection1==’y ‘ ) || ( selection1==’Y ‘ ) )

{

clrscr ( ) ;

attention deficit disorder ( ) ;

}

else if ( ( selection1==’n ‘ ) || ( selection1==’N ‘ ) )

{

return ;

}

}

else

{

clrscr ( ) ;

printf ( “ phone book contacts \*\*” ) ;

for ( num= 0 ; num & lt ; entry ; num++ )

{

printf ( “[ % vitamin D ] . Name: % s % s” , num+1, bbook [ num ] . firstname, bbook [ num ] . lastname ) ;

printf ( “ Home Number: % s” , bbook [ num ] . homenum ) ;

printf ( “ Mobile figure: % s” , bbook [ num ] . mobilenum ) ;

}

printf ( “

Press any key to goto Main Menu: ” ) ;

getch ( ) ;

return ;

}

}

null hunt ( )

{

char uname [ 50 ] ;

char unumber [ 10 ] ;

char firstchar ;

char ccopy [ 50 ] ;

char fchar ;

int I ;

int selection4 ;

int found= F ;

clrscr ( ) ;

printf ( “ \*\* SEARCH MENU \*\*” ) ;

printf ( “

If you want to Search By: ” ) ;

printf ( “ First missive Imperativeness 1 ” ) ;

printf ( “ First name Press 2 ” ) ;

printf ( “ Exit imperativeness 3 ” ) ;

printf ( “

Enter the Choice: “ ) ;

scanf ( “ % vitamin D ” , & A ; selection4 ) ;

switch ( selection4 )

{

instance 1:

clrscr ( ) ;

printf ( “ SEARCH BY FIRST LETTER OF THE NAME” ) ;

printf ( “

Enter First Letter to Search: “ ) ;

fflush ( stdin ) ;

scanf ( “ % degree Celsius ” , & A ; firstchar ) ;

fflush ( stdin ) ;

for ( i= 0 ; i & lt ; entry ; i++ )

{

fchar= bbook [ I ] . firstname [ 0 ] ;

if ( firstchar== fchar )

{

printf ( “ Name: % s % s % s ” , bbook [ I ] . firstname, ” “ , bbook [ I ] . lastname ) ;

printf ( “ Home Number: % s ” , bbook [ I ] . homenum ) ;

printf ( “ Mobile Number: % s ” , bbook [ I ] . mobilenum ) ;

found= T ;

}

}

if ( found== F )

{

clrscr ( ) ;

printf ( “ \*\* DATA NOT FOUND! ! ! \*\*” ) ;

}

printf ( “

Press any cardinal to return to chief Menu ” ) ;

getch ( ) ;

clrscr ( ) ;

bill of fare ( ) ;

interruption ;

instance 2:

clrscr ( ) ;

printf ( “ SEARCH BY FIRST NAME” ) ;

printf ( “

Enter First Name to Search: “ ) ;

fflush ( stdin ) ;

gets ( uname ) ;

fflush ( stdin ) ;

clrscr ( ) ;

for ( i= 0 ; i & lt ; entry ; i++ )

{

if ( strcmp ( uname, bbook [ I ] . firstname ) == 0 )

{

printf ( “ Name: % s % s % s ” , bbook [ I ] . firstname, ” “ , bbook [ I ] . lastname ) ;

printf ( “ Home Number: % s ” , bbook [ I ] . homenum ) ;

printf ( “ Mobile Number: % s

” , bbook [ I ] . mobilenum ) ;

found= T ;

}

}

if ( found== F )

{

clrscr ( ) ;

printf ( “ \*\* DATA NOT FOUND! ! ! \*\*” ) ;

}

printf ( “

Press any cardinal to return to chief Menu ” ) ;

getch ( ) ;

clrscr ( ) ;

bill of fare ( ) ;

interruption ;

instance 3:

abort ( ) ;

interruption ;

default:

clrscr ( ) ;

printf ( “ Invalid Number” ) ;

printf ( “ Press any cardinal to return to seek Menu ” ) ;

getch ( ) ;

interruption ;

}

}