**DESCRIPTION**

* **INTRODUCTION**

This project is on phonebook. The project provides features like password protection, view contact, delete contact, edit contact, speed-dial, calling, etc. the additional feature is taking a backup of all your contact into file which can be used on your smartphone.

* **FUNCTION CALLS AND PROCEDURES**

1. **GRAPHICS MODE.**

grapmode proc ;for graphic mode

mov al,13h

mov ah,00

int 10h

ret

endp

1. **CREATE CONTACT.**

cr\_file: call grapmode

lea dx, msg2 ; module for creating a file

call disp1

call read1 ; read name of file to be

lea dx, buffer1[2] ; created

mov cx, 0

mov ah, 3ch ; create the file

int 21h

mov handle,ax ; push file handle onto stack.

writex: lea dx, msg28 ; ask if data is to

call disp1

mov bx,handle ; retrieve file handle from stack.

mov buffer1[1], 0

write : call readch ; read data character by character.

cmp al,40h

jnc writex

mov buffer1[0], al

cmp buffer1[0], 27 ; check if character is 'Esc'(stop).

jz no

cmp buffer1[0], 0dh

jne neol

lea dx, msg31

call disp1

mov si, dx

mov byte ptr ds:[si + 2], 0

mov cx, 3

jmp com

neol : mov cx, 1

lea dx, buffer1[0]

com : mov ah, 40h ; write to the file

int 21h

mov byte ptr ds:[si + 2], '$'

jmp write

no : mov bx,handle

mov AH,3Eh

int 21h

lea dx, msg16 ; creation successful

call disp1

mov bl,02

call delay

jmp begin

1. **VIEW CONTACT.**

vw\_file:lea dx, msg5 ; module to view the

call disp1 ; contents of a file

lea dx, buffer1[2]

call disp1

lea dx, msg6 ;Number: string

call disp1

lea dx, buffer1[2]

mov al, 02h ;read/write mode

mov ah, 3dh ; open the existing file

int 21h

mov buffer2[0], 0

cmp ax, 2 ; error if file not found

jnz v\_err

lea dx, msg14

call disp1

jmp endv

v\_err: cmp ax, 3 ; error if path not found

jnz cont2

lea dx, msg21

call disp1

mov flag,1

jmp endv

cont2: mov handle, ax

mov bx, handle ;file handle

mov cx, 1 ;no of byte to read

lea dx, buffer1

mov ah, 3fh ; read the file

int 21h

cmp ax, 0 ; stop if end-of-file

jz endv

cmp buffer1[0], 0dh

jnz show

inc buffer2[0]

cmp buffer2[0], 23 ; check if end of page

jnz show

lea dx, msg26

call disp1

show : mov buffer1[1], '$'

lea dx, buffer1

call disp1

mov ax, handle

jmp cont2

end :

jay: cmp flag,0

jne jj2

jmp calling

jj2: jmp optn2

1. **EDIT CONTACT.**

ed\_file:call grapmode

lea dx, buffer1[2] ; created

mov cx, 0

mov ah, 3ch ; create the file

int 21h

mov handle, ax ; push file handle onto stack.

writex2:lea dx, msg28 ; ask if data is to

call disp1

mov bx, handle ; retrieve file handle from stack.

mov buffer1[1], 0

write2 :call readch ; read data character by character.

cmp al,40h

jnc writex2

mov buffer1[0], al

cmp buffer1[0], 27 ; check if character is 'Esc'(stop).

jz no2

cmp buffer1[0], 0dh

jne neol2

lea dx, msg31

call disp1

mov si, dx

mov byte ptr ds:[si + 2], 0

mov cx, 3

jmp com2

neol2 :mov cx, 1

lea dx, buffer1[0]

com2 :mov ah, 40h ; write to the file

int 21h

mov byte ptr ds:[si + 2], '$'

jmp write2

no2 :mov bx,handle

mov AH,3Eh

int 21h

lea dx, msg16 ; creation successful

call disp1

mov bl,02

call delay

jmp begin

1. **DELETE CONTACT.**

dl\_file:call grapmode

lea dx, buffer1[2]

mov ah, 41h ; delete the file

int 21h

cmp ax, 2 ; error if file not found

jnz err2

lea dx, msg14

call disp1

jmp endd

err2 : cmp ax, 5 ; error if access denied

jnz done

lea dx, msg15

call disp1

jmp endd

done : lea dx, msg17 ; delete successful

call disp1

endd : mov bl,02

call delay

jmp begin

1. **CREATE BUTTON.**

mov cx,0bh ;column

mov dx,0b0h ;row

mov al,0fh ;button colour(white)

mov ah,0ch ;change colour for single pixel

sna: int 10h

inc cx

cmp cx,4ah

jne sna

inc dx

mov cx,0Bh

cmp dx,0c2h

jne sna

1. **STRING ON BUTTON.**

mov ax,data

mov es,ax

mov al,01h ;string write mode with attr.

mov bh,0h ;page no

mov bl, 0fh ;colour

mov cx,06h ;string length

mov dl,2h ; 1Bh+4h=20h=32d 32d/8d=4pixel

mov dh,17h

mov bp,offset but1

mov ah,13h ;display string with attr. in graphical mode

int 10h

1. **MOUSE CLICK.**

again: mov ax,0000h

int 33h

cmp ax,0000h

je again ;End Mouse initialization....

;show initialized mouse pointer

mov ax, 0001h

int 33h ;endsssssssss show initialized mouse pointer

;check for button click..............

check: mov fileno,0h

mov ax,0003h

int 33h

cmp bx,1h ;any button pressed

jne check

shr cx,01h

cmp cx,0Bh ;greater than or equal 0B we want!!!!

jc che1

cmp cx,4ah ;less than 4A!!!!

jnc che1

cmp dx,0b0h ;greater than or equal 0B0 we want!!!!

jc che1

cmp dx,0c3h ;less than C3!!!!

jnc che1

jmp cr\_file ;jump to create

1. **DISPLAY STRING ON SCREEN.**

disp1 proc

mov ah, 09h ; module for display of

int 21h ; a string on screen

ret

disp1 endp

1. **READING STRING FROM KEYBOARD.**

read1 proc ; module for reading

mov buffer1[0], 80

mov buffer1[1],0

lea dx, buffer1

mov ah, 0ah ; read string from keyboard

int 21h

mov bl, buffer1[1]

mov bh, 0

add bx, 2

mov buffer1[bx], 0 ; ASCIIZ string, so

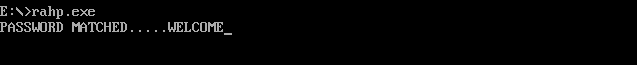
ret ; terminate with 0

read1 endp

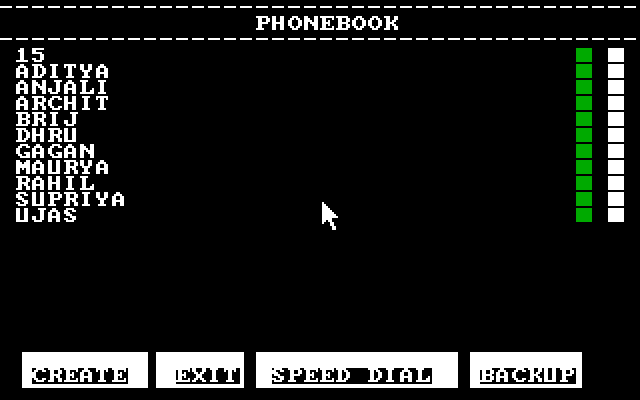
**SCREENSHOTS**

* **PASSWORD PROTECTION**



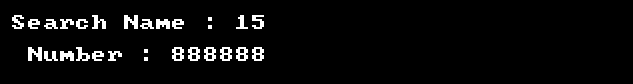


* **HOME PAGE**

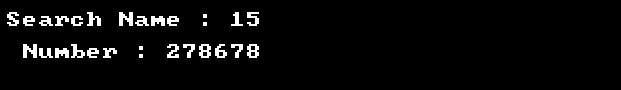


* **EDIT CONTACT**

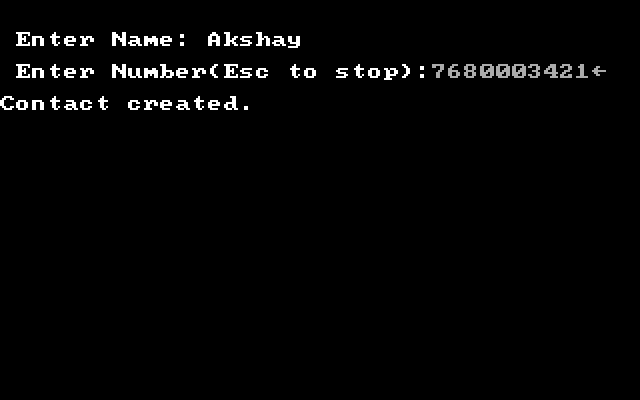
(ORIGINAL CONTACT)



(EDITTED CONTACT)



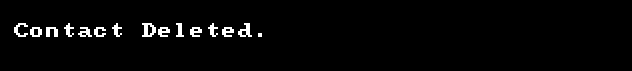
* **CREATE CONTACT**



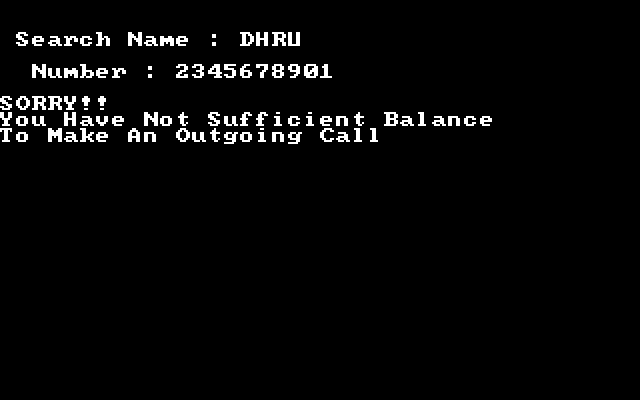
* **VIEW CONTACT**



* **DELETE CONTACT**



* **CALL CONTACT**



* **SPEED DIAL**

