//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

// lab 1

// was written by : [Reham Abass 201277811] and [Byata 205797475]

#include<dos.h>

#include<stdio.h>

//////////////////////////////////////////////////

void interrupt (\*old9)(void);

void interrupt(\*old8)(void);

//////////////////////////////////////////////

volatile char \* password; //genral variable to save real password

volatile long generalcounter; //claculate time from calling the function until it finishes

volatile int counter; //counter between each press

volatile int keyTimer; //global variable for the keyTime- max time between each press\*19

volatile int size; //size of the real password

volatile int index; //index inside the password

volatile int flag; //flag to compare pressed and released

volatile long timer; //for use in my8

volatile int index=0;

char letter1, letter2;

//////////////////////////////////////////////

void changePassword() {

int i;

for (i = 0; i < size; i++) {

if (password[i] == 'z') password[i] = 'a';

else

password[i] = password[i] + 1;

}

}

///////////////////////////////////////////////

void interrupt my9(){ //keyboard interrupt

flag++;

if (flag%2 ==0) { //pressed

counter = 0; //if pressed so counter=0

}

old9(); // call old routine

}

//////////////////////////////////////////////

//////////////////////////////////////////////

void interrupt my8(){ //time interrupt

if (generalcounter == 0) timer = 0;

timer = generalcounter / 19;

if ((generalcounter %19 == 0)&&(timer != 0) && (timer % 15 == 0)) {

changePassword();

printf("\nI have changed the password");

}

generalcounter++;

if ((counter == keyTimer-1)){

printf("\nstart again(passed 5 seconds) ");

index=0; //update the index to the begining free the input

counter=0;

}

else

counter++;

old8(); // call old routine

}

////////////////////////////////////////

void start(){

old8 = getvect(8); //save old vector

old9 =getvect(9); //save old vector

setvect(8,my8); //set my function 8

setvect(9,my9); //set my function 9

counter=0; //initialization

generalcounter=0; //initialization

flag=0;

}

/////////////////////////////////////////

void end(){

setvect(8,old8);

setvect(9,old9);

}

//////////////////////////////////////////

long silence\_keyboard(char \*pass, int size1, int keyTime){

index = 0; //index points on first letter

keyTimer = keyTime\*19; // copy from local variable to general variable

size = size1-1; // copy from local variable to general variable

password = pass; // copy from local variable to general variable

start(); //initializations

while(index < size){

letter1 = password[index];

asm{

mov ah,1

INT 16h

JZ info

jmp end1

}

info:

asm{

mov AH,0

INT 16h

mov letter2,al

}

if(letter2 == '1') {printf("\npressed 1 time is = %d",timer);}

else if(letter1 == letter2){index++; printf("\n correct letter = %c",letter2);}

else {index=0;printf("\nstart again - uncorrect char");counter=0;}

end1:

}//while

end(); //restore vectors

return timer;

}

///////////////////////////////////////////

void main(void){

char str[] ="aabbcac"; //the password

int keyTime = 5; //timer between each press

long time = silence\_keyboard(str, sizeof(str), keyTime);//call the function which return time until find the right password

printf("\ntotal time : %ld\n", time);

}

//////////////////////////////////////////////