

ORG 0000H

-----

MOV A,#38H ;select

ACALL LCD\_COMMAND

MOV A,#0EH ;display on cursor on

ACALL LCD\_COMMAND

MOV A,#01H ;clear display

ACALL LCD\_COMMAND

-----

MOV 55H,#01H ; 1st digit,value 35h

MOV 56H,#05H ; 2nd digit,value 36h

MOV 57H,#09H ; 3rd digit,value 37h

MOV R4,#00H ; TOTAL ATTEMPT COUNTER;to trigger security breaCH(40H)

-----START-----

MOV P0, #11111111B ;7 segment

MOV R5, #00H ;index

MOV R6, #23 ;scroll

scroll:

ACALL delay\_move

DJNZ R6, GO

GO:

MOV A, #80H

ACALL LCD\_COMMAND

MOV DPTR, #String1

```
MOV A, R5
ADD A, DPL
MOV DPL, A
CLR A
ADDC A, DPH
MOV DPH, A
INC R5
MOV R3, #32
back:
CLR A
MOVC A, @A+DPTR
JZ EXIT
ACALL LCD_DATA
INC DPTR
DEC R3
DJNZ R3, back
SJMP scroll

EXIT:
SJMP print

print:
MOV A, #82H
ACALL LCD_COMMAND
MOV DPTR, #String2
back2:
CLR A
```

MOVC A,@A+DPTR

JZ EXIT2

ACALL LCD\_DATA

INC DPTR

SJMP back2

EXIT2:

ACALL delay\_2sec

SJMP print2

print2:

MOV A,#81H

ACALL LCD\_COMMAND

MOV DPTR,#String3

back3:

CLR A

MOVC A,@A+DPTR

JZ EXIT3

ACALL LCD\_DATA

INC DPTR

SJMP back3

EXIT3:

ACALL delay\_2sec

SJMP print3

print3:

```
MOV A,#81H
ACALL LCD_COMMAND
MOV DPTR,#String4
back4:
CLR A
MOVC A,@A+DPTR
JZ LCD_INIT
ACALL LCD_DATA
INC DPTR
SJMP back4
```

```
LCD_INIT:
ACALL delay_2sec
MOV A,#01H
ACALL LCD_COMMAND
SJMP KEYPAD_DISPLAY
```

;

---

```
KEYPAD_DISPLAY:
MOV 40H,#00H    ; TEMPORARY Attempt counter
MOV R2,#00H     ; Input index
```

```
start:
MOV P3,#0FFH

CLR P3.0

JB P3.4,column2_row1
```

MOV A,#01H

ACALL handle

column2\_row1:

JB P3.5,column3\_row1

MOV A,#02H

ACALL handle

column3\_row1:

JB P3.6,column1\_row2

MOV A,#03H

ACALL handle

column1\_row2:

SETB P3.0

CLR P3.1

JB P3.4,column2\_row2

MOV A,#04H

ACALL handle

column2\_row2:

JB P3.5,column3\_row2

MOV A,#05H

ACALL handle

column3\_row2:

JB P3.6,column1\_row3

MOV A,#06H

ACALL handle

column1\_row3:

SETB P3.1

CLR P3.2

JB P3.4,column2\_row3

MOV A,#07H

ACALL handle

column2\_row3:

JB P3.5,column3\_row3

MOV A,#08H

ACALL handle

column3\_row3:

JB P3.6,column1\_row4

MOV A,#09H

ACALL handle

column1\_row4:

SETB P3.2

CLR P3.3

JB P3.4,column2\_row4

MOV A,#0AH

INC R0

ACALL handle

column2\_row4:

JB P3.5,column3\_row4

MOV A,#00H

ACALL handle

column3\_row4:

JB P3.6, J

MOV A,#0BH

ACALL handle

J:LJMP start

;

---

handle:

MOV R7,A

CJNE A, #0BH, store

ACALL check

RET

store:

INC R2

SJMP store1

store1:

CJNE R2, #01H, store2

MOV 35H, A

```
ACALL display  
MOV A,#80H  
ACALL LCD_COMMAND  
MOV A,35H  
ADD A,#30H  
ACALL LCD_DATA  
ACALL delay  
RET
```

```
store2:  
CJNE R2, #02H, store3  
MOV 36H, A  
ACALL display  
MOV A,#81H  
ACALL LCD_COMMAND  
MOV A,36H  
ADD A,#30H  
ACALL LCD_DATA  
ACALL delay  
RET
```

```
store3:  
MOV 37H, A  
MOV R2, #00H  
ACALL display  
MOV A,#82H  
ACALL LCD_COMMAND
```



```
MOV A,37H
ADD A,#30H
ACALL LCD_DATA
ACALL delay
MOV A,#01H
ACALL LCD_COMMAND
RET
```

check:

```
MOV A,35H
XRL A,#03H
JNZ check2
ACALL delay
MOV A,36H
XRL A,#05H
JNZ check2
ACALL delay
MOV A,37H
XRL A,#09H
JNZ check2
ACALL delay
ACALL admin_mode
RET
```

check2:

```
MOV A,35H
MOV B,55H
```

XRL A,B

JNZ denied

ACALL delay

MOV A,36H

MOV B,56H

XRL A,B

JNZ denied

ACALL delay

MOV A,37H

MOV B,57H

XRL A,B

JNZ denied

ACALL delay

ACALL granted

RET

denied:

MOV R2,#00H

INC R4 ;WRONG ATTEMPS IN WHOLE LIFE

INC 40H ;WRONG ATTEMPTS FOR SECURITY BREACH

MOV A,40H

XRL A,#3

JZ security\_breach

MOV A, #81H

ACALL LCD\_COMMAND

MOV DPTR, #String5

denied\_back:

CLR A

MOVC A, @A+DPTR

JZ ex

ACALL LCD\_DATA

INC DPTR

SJMP denied\_back

granted:

MOV R2,#00H

MOV 40H,#0

MOV A, #81H

ACALL LCD\_COMMAND

MOV DPTR, #String6

granted\_back:

CLR A

MOVC A, @A+DPTR

JZ ex

ACALL LCD\_DATA

INC DPTR

SJMP granted\_back

ex:

MOV A, #01H

ACALL LCD\_COMMAND

LJMP start

;-----7segment-----

display:

MOV DPTR, #table

MOV A, R7

MOVC A, @A+DPTR

MOV P0, A

RET

;-----SECURITY

BREACH-----

security\_breach:

MOV 40H,#0

MOV DPTR, #String7

security\_breach\_back:

CLR A

MOVC A, @A+DPTR

JZ BUZZER

ACALL LCD\_DATA

INC DPTR

SJMP security\_breach\_back

BUZZER:

CLR P2.5

MOV B, #40

LOOP:

SETB P2.4

ACALL delay

CLR P2.4

ACALL delay

DJNZ B,LOOP

SJMP DISABLE\_KEYPAD

DISABLE\_KEYPAD:

MOV P3,#0FFH

ACALL delay\_2sec

ACALL delay\_2sec

ACALL delay\_2sec

ACALL delay\_2sec

ACALL delay\_2sec

MOV A,#01H

ACALL LCD\_COMMAND

LJMP KEYPAD\_DISPLAY

;-----

ADMINMODE-----

admin\_mode:

MOV R2,#00H

MOV R7,#00H ;USED FOR DISPLAY OF MODE ON LCD

MOV DPTR, #String8

admin\_mode\_back:

CLR A

MOVC A, @A+DPTR

JZ next

ACALL LCD\_DATA

INC DPTR

SJMP admin\_mode\_back

next:

```
ACALL delay_2sec  
  
MOV A,#01H  
  
ACALL LCD_COMMAND  
  
MOV A,#80H  
  
ACALL LCD_COMMAND  
  
MOV DPTR,#String9  
  
next_back:  
  
CLR A  
  
MOVC A,@A+DPTR  
  
JZ T  
  
ACALL LCD_DATA  
  
INC DPTR  
  
SJMP next_back
```

```
T:LJMP SELECT_SCAN
```

```
SELECT:  
  
MOV R7,A  
  
MOV A,#89H  
  
ACALL LCD_COMMAND  
  
MOV A,R7  
  
ADD A,#30H  
  
ACALL LCD_DATA  
  
ACALL delay  
  
MOV A,#01H  
  
ACALL LCD_COMMAND  
  
MOV A,R7
```

XRL A,#01H

JZ MODE1

MOV A,R7

XRL A,#02H

JZ K

MOV A,R7

XRL A,#03H

JZ H

ACALL EXIT\_ADMINMODE

K:LJMP MODE2

H:LJMP MODE3

;-----MODE1-----

MODE1:

MOV R2,#00H ;INDEXING

MOV A,#01H

ACALL LCD\_COMMAND

MOV A,#80H

ACALL LCD\_COMMAND

MOV DPTR,#String11

MODE1\_BACK:

CLR A

MOVC A,@A+DPTR

JZ CLEARNGO

ACALL LCD\_DATA

INC DPTR

SJMP MODE1\_BACK

CLEARNGO:

MOV A,#01H

ACALL LCD\_COMMAND

LJMP SCAN\_PASSWORD

;

---

STORE\_PASSWORD:

MOV R7,A ;7SEGMENT DISPLAY

INC R2

SJMP pass\_store1

pass\_store1:

CJNE R2, #01H,pass\_store2

MOV 55H, A

ACALL display

MOV A,#80H

ACALL LCD\_COMMAND

MOV A,55H

ADD A,#30H

ACALL LCD\_DATA

ACALL delay

RET

pass\_store2:

CJNE R2, #02H,pass\_store3

MOV 56H, A



```
ACALL display  
MOV A,#81H  
ACALL LCD_COMMAND  
MOV A,56H  
ADD A,#30H  
ACALL LCD_DATA  
ACALL delay  
RET
```

```
pass_store3:  
MOV 57H, A  
MOV R2, #00H  
ACALL display  
MOV A,#82H  
ACALL LCD_COMMAND  
MOV A,57H  
ADD A,#30H  
ACALL LCD_DATA  
ACALL delay  
MOV A,#01H  
ACALL LCD_COMMAND  
MOV A,#80H  
ACALL LCD_COMMAND  
MOV DPTR,#String14  
PASS_SET_BACK:  
CLR A  
MOVC A,@A+DPTR
```

```
JZ TEXTCLEAR  
  
ACALL LCD_DATA  
  
INC DPTR  
  
SJMP PASS_SET_BACK
```

```
TEXTCLEAR:  
  
MOV A,#01H  
  
ACALL LCD_COMMAND  
  
LJMP next
```

```
;-----MODE2-----
```

```
MODE2:  
  
MOV A,#01H  
  
ACALL LCD_COMMAND  
  
MOV A,#80H  
  
ACALL LCD_COMMAND  
  
MOV DPTR,#String12  
  
MODE2_BACK:  
  
CLR A  
  
MOVC A,@A+DPTR  
  
JZ NOA  
  
ACALL LCD_DATA  
  
INC DPTR  
  
SJMP MODE2_BACK
```

```
NOA: ;ASSUMING MAX UNIVERSAL WRONG ATTEMPT TO BE 99  
  
MOV A,#8BH
```

```

ACALL LCD_COMMAND

MOV A,R4

MOV B,#10      ;A,QUOTIENT , B ,REMAINDER

DIV AB

ADD A,#30H

ACALL LCD_DATA

MOV A,#8CH

ACALL LCD_COMMAND

MOV A,B

ADD A,#30H

ACALL LCD_DATA

ACALL delay_2sec

MOV A,#01H

ACALL LCD_COMMAND

LJMP next

```

```

;-----MODE3-----

```

```

MODE3:

MOV R4,#00H

MOV A,#01H

ACALL LCD_COMMAND

MOV A,#80H

ACALL LCD_COMMAND

MOV DPTR,#String13

MODE3_BACK:

CLR A

MOVC A,@A+DPTR

JZ EXIT_MODE3

```

ACALL LCD\_DATA

INC DPTR

SJMP MODE3\_BACK

EXIT\_MODE3:

ACALL delay\_2sec

MOV A,#01H

ACALL LCD\_COMMAND

LJMP next

;

---

SCAN\_PASSWORD:

MOV P3,#0FFH

CLR P3.0

JB P3.4,c2r1

MOV A,#01H

ACALL STORE\_PASSWORD

c2r1:

JB P3.5,c3r1

MOV A,#02H

ACALL STORE\_PASSWORD

c3r1:

JB P3.6,c1r2

MOV A,#03H

ACALL STORE\_PASSWORD

c1r2:

SETB P3.0

CLR P3.1

JB P3.4,c2r2

MOV A,#04H

ACALL STORE\_PASSWORD

c2r2:

JB P3.5,c3r2

MOV A,#05H

ACALL STORE\_PASSWORD

c3r2:

JB P3.6,c1r3

MOV A,#06H

ACALL STORE\_PASSWORD

c1r3:

SETB P3.1

CLR P3.2

JB P3.4,c2r3

MOV A,#07H

ACALL STORE\_PASSWORD

c2r3:

JB P3.5,c3r3

MOV A,#08H

ACALL STORE\_PASSWORD

c3r3:

JB P3.6,c1r4

MOV A,#09H

ACALL STORE\_PASSWORD

c1r4:

SETB P3.2

CLR P3.3

JB P3.4,c2r4

MOV A,#0AH

INC R0

ACALL STORE\_PASSWORD

c2r4:

JB P3.5,c3r4

MOV A,#00H

ACALL STORE\_PASSWORD

c3r4:

JB P3.6,EFG

MOV A,#0BH

ACALL STORE\_PASSWORD

EFG:

LJMP SCAN\_PASSWORD

;

---

SELECT\_SCAN:

MOV P3,#0FFH

CLR P3.0

JB P3.4,c2\_r1

MOV A,#01H

ACALL SELECT

c2\_r1:

JB P3.5,c3\_r1

MOV A,#02H

ACALL SELECT

c3\_r1:

JB P3.6,c1\_r2

MOV A,#03H

ACALL SELECT

c1\_r2:

SETB P3.0

CLR P3.1

JB P3.4,c2\_r2

MOV A,#04H

ACALL SELECT

c2\_r2:

JB P3.5,c3\_r2

MOV A,#05H

ACALL SELECT

c3\_r2:

JB P3.6,c1\_r3

MOV A,#06H

ACALL SELECT

c1\_r3:

SETB P3.1

CLR P3.2

JB P3.4,c2\_r3

MOV A,#07H

ACALL SELECT

c2\_r3:

JB P3.5,c3\_r3

MOV A,#08H

ACALL SELECT

c3\_r3:

JB P3.6,c1\_r4

MOV A,#09H



ACALL SELECT

c1\_r4:

SETB P3.2

CLR P3.3

JB P3.4,c2\_r4

MOV A,#0AH

INC R0

ACALL SELECT

c2\_r4:

JB P3.5,c3\_r4

MOV A,#00H

ACALL SELECT

c3\_r4:

JB P3.6, G

MOV A,#0BH

ACALL SELECT

G: LJMP SELECT\_SCAN

;

---

EXIT\_ADMINMODE:

MOV A,#81H

ACALL LCD\_COMMAND

MOV DPTR,#String10

EXIT\_ADMINMODE\_BACK:

CLR A

MOVC A,@A+DPTR

JZ TARGETOUTOFRANGE

ACALL LCD\_DATA

INC DPTR

SJMP EXIT\_ADMINMODE\_BACK

TARGETOUTOFRANGE:

MOV A,#01H

ACALL LCD\_COMMAND

LJMP print3

;

---

LCD\_COMMAND:

ACALL delay

MOV P1, A

CLR P2.1

SETB P2.3

CLR P2.3

RET

LCD\_DATA:

ACALL delay

MOV P1, A

SETB P2.1

SETB P2.3

CLR P2.3

RET

delay:

MOV R0, #0FFH

L2: MOV R1, #0FFH

L3: DJNZ R1, L3

DJNZ R0, L2

RET

delay\_move:

MOV R0, #05H

L1: DJNZ R0, L1

RET

delay\_2sec:

MOV R3, #200

OUTER\_LOOP:

MOV R2, #250

MIDDLE\_LOOP:

MOV R1, #40

INNER\_LOOP:

DJNZ R1, INNER\_LOOP

DJNZ R2, MIDDLE\_LOOP

DJNZ R3, OUTER\_LOOP

RET

table:

DB 1000000B ;0

DB 1111001B ;1

DB 0100100B ;2

DB 0110000B ;3

DB 0011001B ;4

DB 0010010B ;5

DB 0000010B ;6

DB 1111000B ;7

DB 0000000B ;8

DB 0010000B ;9

DB 0111111B ;A

DB 0111111B ;B

String1: DB "WELCOME     INITIALISING ", 0

String2: DB ".....", 0

String3: DB "SAFETY CONSOLE", 0

String4: DB "ENTER PASSWORD", 0

String5: DB "ACCESS DENIED", 0

String6: DB "ACCESS GRANTED", 0

String7: DB "SYSTEM LOCKED", 0

String8: DB "ADMIN MODE",0

String9: DB "PINMODE:",0

String10: DB "EXIT ADMIN MODE",0

String11: DB "SET PASSWORD",0

String12: DB "X ATTEMPTS ",0

String13: DB "RESET ATTEMPTS",0

String14: DB "PASSWORD STORED",0

END