BINARY SEARCH

```
.MODEL SMALL
;M ; MACRO TO DISPLAY THE MESSAGE.
DISPLAY MACRO MSG
 LEA DX, MSG
 MOV AH, 09H
 INT 21H
 END M
.DATA
LIST DB 01H, 05H, 07H, 10H, 12H, 14H.
NUMBER EQU ($-LIST) ; NUMBER HAS THE VALUE 6
KEY DB 10H
MSG1 DB 0DH, 0AH, "ELEMENT FOUND IN THE LIST... $"
MSG2 DB 0DH, 0AH, "SEARCH FAILED! ELEMENT NOT FOUND IN THE
                   LIST" $"

.CODE
START : MOV AX, @DATA
        MOV DS, AX
        MOV CH, NUMBER-1 ; HIGH VALUE (5) (6-1=5)
        MOV CL, 00H ; LOW VALUE

AGAIN : MOV SI, OFFSET LIST ; LEA SI, LIST
        XOR AX, AX ; MOV AX, 00H
        CMP CL, CH ; SUB OF CL-CH
        JE NEXT
        JNC CARRY FAILED

NEXT : MOV AL, CL ; AL= 00H
       ADD AL, CH ; AL = 00+05 = 05
       SHR, AL, 01H ; DIVIDE BY 2
       MOV BL, AL ; BL → INDEX OF MIDDLE ELE
       XOR AH, AH ; CLEAR AH.
```

```
        MOV BP, AX
        MOV AL, DS: [BP][SI]
        CMP AL, KEY     ; COMPARE KEY & A[I]
        JE SUCCESS      ; IF EQUAL, DISPAY SUCCESS MESSAGE
        JC INCLOW
        MOV CH, BL      ; IF KEY > A[I]  SHIFT HIGH
        DEC CH          ; CH WILL HAVE INDEX OF MID-1 ELEMENT
        JMP AGAIN

INCLOW: MOV CL, BL      ; IF KEY< A[I]  SHIFT LOW
        INC CL          ; ~~SHIFT~~ CL → INDEX OF MID+1 ELEMENT
        JMP AGAIN

SUCCESS : DISPLAY & MSG1
        JMP FINAL

FAILED :  DISPLAY MSG2

FINAL :   MOV AH, 4CH
        INT 21H

END START
```