* Binary Search:

```
.MODEL SMALL
; Macro to Display Message

DISPLAY MACRO MSG
        LEA DX, MSG
        MOV AH, 09H
        INT 21H

ENDM

.DATA
LIST DB 01H, 05H, 07H, 10H, 12H, 14H
NUMBER EQU ($ - LIST)

KEY DB 012H

MSG1 DB 0DH, 0AH, "ELEMENT FOUND IN THE
                   LIST... $"

MSG2 DB 0DH, 0AH, " SEARCH FAILED!!
                   ELEMENT NOT FOUND "

.CODE

START : MOV AX, @DATA
        MOV DS, AX
```

```
        MOV  CH, NUMBER-1        ; High value.
        MOV  CL, 00H             ; low value

AGAIN : MOV  SI , OFFSET LIST

        XOR  AX, AX
        CMP  CL, CH              ; ck ch  cl=ch  cl>ch
        JE   NEXT                  cf=1  zf=1  cf=0
        JNC  FAILED

NEXT:   MOV  AL, CL
        ADD  AL, CH
        SHR  AL, 01H             ; ∴ Divide By 2
        MOV  BL, AL
        XOR  AH, AH              ; Clear AH
        MOV  BP, AX
        MOV  AL, DS : [BP][SI]

        CMP  AL, KEY             ; compare Key
                                   and A[I]

        JE   SUCESS              ; If equal, Display
        JC   INCLOW                success message.

        MOV  CH, BL
        DEC  CH                  ; If KEY > A[I]
                                   Shift High

        JMP  AGAIN
```

```
INCLOW : MOV CL, BL
         INC CL
         JMP AGAIN

SUCCESS : DISPLAY MSG1
          JMP FINAL

FAILED : DISPLAY MSG2

FINAL : MOV AH, 4CH
        INT 21H
```

; If KEY<API]
  Shift low

; JOB OVER
  TERMINATE