

Lab Program

1) Binary Search

• model small

; macro to display the 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) ; number here is having the value 6
```

```
key db 10H
```

```
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 ... 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 ; subtraction of cl - ch
```

```
        je next
```

```
        jnc failed
```

```
next : mov al, cl ; al = 00h
```

add al, ch ; $al = 00 + 05 = 05$
shr al, 01h ; divide by 2 (al will have index middle element)

mov bl, al ; bl \rightarrow index of middle element

xor ah, ah ; clear ah

mov bp, ax

mov al, ds: [bp][si]

cmp al, key ; compare key and a[i]

je success ; if equal, display success message

jc inflow

mov ch, bl ; if $key > a[i]$ shift high

dec ch ; ch will have index of middle-1 element (search from low to mid-1)

jmp again

inflow : mov cl, bl ; if $key < a[i]$ shift low

inc cl ; cl will have index of middle+1 element (search from mid+1 to high)

jmp again

success : display msg₁

jmp final

failed : display msg₂

final : mov ah, 4ch

int 21h

end start