**MP ; programs**

1. **Binary search:**

**.model small**

**.data**

ARRAY DW 1234H, 2345H, 3356H

LENG DW ($-ARRAY)/2 ; NO OF ELEMENTS IN THE ARRAY

SUCMSG DB 10, 13, ’SUCESSFUL SEARCH, ELEMENT IS FOUND AT THE POSITION’

POS DB ‘ $’

UNSUCMSG DB 10, 13 , ‘UNSUCESSFUL SEARCH$’

SRCHKEY DW 1234H

.CODE

MOV AX,@DATA

MOV DS, AX

MOV BX, 00H ;BX = LOW

MOV DX, LENG ; DX = HIGH

MOV CX, SRCHKEY

AGAIN: CMP BX, DX

JA FAILURE ; IF BX > DX WHILE(LOW>HIGH)

ADD BX, DX ;LOW+HIGH

MOV AX, BX

SHR AX, 1 ;MID = (LOW+HIGH)/2

MOV SI, AX ; POINTING SI TO MID ELE

DEC SI

ADD SI, SI ; SI=2\*(AX) ,SINCE IT’S A WORD OPERATION

CMP CX, ARRAY[SI] ; COMPARING SEARCH KEY WITH A[MID]

JAE NEXT

DEC AX ; HIGH=MID-1

MOV DX, AX

JMP AGAIN

NEXT: JE SUCCESS

INC AX ; LOW=MID+1

MOV BX, AX

JMP AGAIN

SUCCESS: ADD AL, ’0’ ;ADD AL,30H ASCII CONVERSION

MOV POS, AL

LEA DX, SUCMSG

JMP DISP

FAILURE: LEA DX , UNSUCMSG

DISP: MOV AH,09H

INT 21H

MOV AH , 4CH

INT 21H

END

1A) .MODEL SMALL

;Macros to be defined after **.MODEL**

DISPLAY MACRO MSG ;THEN **.DATA** STARTS

LEA DX, MSG ; THEN **.CODE**

MOV AH,09H

INT 21H

ENDM

INITDS MACRO

MOV AX,@DATA

MOV DS,AX

ENDM

EXIT MACRO

MOV AH,4CH

INT 21H

ENDM

1. **Bubble sort**

.model small

.data

Org 1000h

num db 01,02,05,03,04

count equ o5h ; count db ($-num)/2

.code

Initds

mov dx, count-1

outer: mov cx,dx

lea si,num

inner: mov al,[si]

cmp al,[si+1]

jbe nochgh

xchg [si+1],al

mov [si],al

nochge: inc si

loop inner

dec dx

jnz outer

exit

end

1. To display the current time of the system

.MODEL SMALL

.DATA

MSG DB 10,13, ‘THE CURRENT TIME IS --- $’

TIME DB ’00:00:00’

.CODE

INITDS

LEA BX,TIME

LEA DX,MSG

DISPLAY MSG

CALL GET\_TIME

DISPLAY TIME

EXIT

GET\_TIME PROC

MOV AH, 2CH

INT 21H

MOV AL, CH

CALL CONVERT

MOV [BX], AX

MOV AL, CL

CALL CONVERT

MOV [BX+3], AX

MOV AL, DH

CALL CONVERT

MOV [BX+6], AX

RET

GET\_TIME ENDP

CONVERT PROC

MOV AH , 0

MOV DL, 10D

DIV DL

OR AX, 3030H

RET

CONVERT ENDP

END

1. Pallindrome or NOT

.MODEL SMALL

.DATA

ORG 2000H

STR DB 40 DUP (“$”)

REV DB 40 DUP (“$”)

MSG1 DB 10,13, ‘ ENTER THE STRING$’

MSG2 DB 10,13, ‘ THE REVERSE STRING IS: $’

PAL DB 10,13, ‘ THIS STRING IS A PALINDROME$’

NPAL DB 10,13 ‘THIS STRING IS NOT A PALINDROME$’

.CODE

INITDS

MOV ES, AX

DISPLAY MSG1

MOV CX,00H ;XOR CX,CX

LEA SI, STR

READ: MOV AH,01H

INT 21H

CMP AL,ODH

JZ NEXT

NEXT : LEA DI,REV

MOV BP, CX ;SAVING THE COUNTER VALUE IN SOME REG

TRANS: DEC SI

MOV AL,[SI]

MOV [DI],AL

INC DI

LOOP TRANS

DISPLAY MSG2

DISPLAY REV

LEA DI,REV

LEA SI,STR

MOV CX,BP

REPE: CMPSB

JNZ FAIL

DISPLAY PAL

JMP EXIT

FAIL: DISPLAY NPAL

EXIT

END