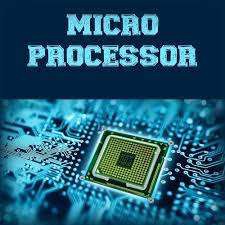
****

## **Atma Ram Sanatan Dharma College**

University of Delhi



**MICROPROCESSOR**

Practical File for Paper Code 32347504

**Submitted By**

Deepali pathania

College Roll No. 21/18076

BSc (Hons) Computer Science

**Submitted To**

Mr Mahesh Bhandari

Department of Computer Science

**Ques 4 :- Write a program for Linear search and Binary Search.**

**Solution:-**

Linear – Search:-

.model small ; contain two segment data and code

.STACK ; tells the assembler to reserve storage

.386 ;instructions for the 80386 processor

.DATA ; start of data segment

ARRAY DB 9 DUP(?) ; Declaring an array with garbage MESS01 DB 13,10,"MAX. NO. OF ELEMENTS IN ARRAY IS 9 $"

MESS02 DB 13,10," $" ; 10 is the ASCII control code for line feed while 13 is the code for carriage return

MESS1 DB 13,10,"ENTER THE NUMBER OF ELEMENTS: $" MESS0 DB 13,10,"ENTER THE NUMBER: $"

MESS2 DB 13,10,"ENTER THE ELEMENT TO BE SEARCHED: $" MESS3 DB 13,10,"VALUE FOUND AT LOCATION - $"

MESS4 DB 13,10,"VALUE NOT FOUND!!!$" ErrMess DB 13,10,"ERROR IN INPUT DIGIT$"

DAT DB ? ; set byte size variable number dw ? ; set double word variable POS DW ? ; set double word variable

.code ; start of code segment

.startup ; Generates program start-up code

MOV DX, OFFSET MESS01 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

MOV DX, OFFSET MESS02 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

MOV DX,OFFSET MESS1 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

MOV AH, 01

INT 21H ; input from user CMP al,39h

JBE abc ; jump if below or equal to

MOV DX, OFFSET ErrMess MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

JMP myexit

abc: AND AL, 0FH

MOV AH, 0

MOV number, AX

MOV CX, AX ; SET COUNTER AL TIMES MOV DI, 0

; INPUT ELEMENTS IN ARRAY

MYLOOP:

MOV DX, OFFSET MESS0 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

; Tens digit

MOV AH, 01

INT 21H ; input from user CMP AL, 39H

JBE abc2 ;; jump if below or equal to

MOV DX, OFFSET ErrMess MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

JMP myexit

abc2: AND al,0fh

SHL AL, 4 ; multiply by 16 MOV BL, AL

; Units digit

MOV AH,01 INT 21H

cmp al,39h

jbe abcx ; jump if below or equal to

MOV DX,OFFSET ErrMess MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 jmp myexit

abcx:

AND al,0fh ADD al, bl

MOV ARRAY[DI], AL INC DI

LOOP MYLOOP

;INPUT ELEMENT TO BE SEARCHED

MOV DX,OFFSET MESS2 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

; Tens digit

MOV AH,01 INT 21H

cmp al,39h

jbe abcl ; jump if below or equal to

MOV DX,OFFSET ErrMess MOV AH,09

INT 21H

jmp myexit

abcl:

and al,0fh

shl al,4 ; multiply by 16 mov bl,al

; Units digit

MOV AH,01

INT 21H ; input from user cmp al,39h

jbe abcm ; jump if below or equal to

MOV DX,OFFSET ErrMess MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 jmp myexit

abcm: and al,0fh

add al,bl mov DAT,AL

; SEARCH PROCESS

MOV AX, DS MOV ES, AX

MOV AL, DAT

CLD ; Auto-Increment Mode MOV CX, number

MOV DI, OFFSET ARRAY REPNE SCASB

CMP CX, 0

JE NOTFOUND

MOV DX, OFFSET MESS02 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

MOV DX, OFFSET MESS3 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

SUB NUMBER, CX

ADD NUMBER,30H ; convert the hexadecimal digits into its equivalent ASCII

MOV DX, NUMBER MOV AH, 02

INT 21H ; Output a character present in DL , as AH value is 2

JMP myexit

NOTFOUND:

MOV DX,OFFSET MESS4 MOV AH,09

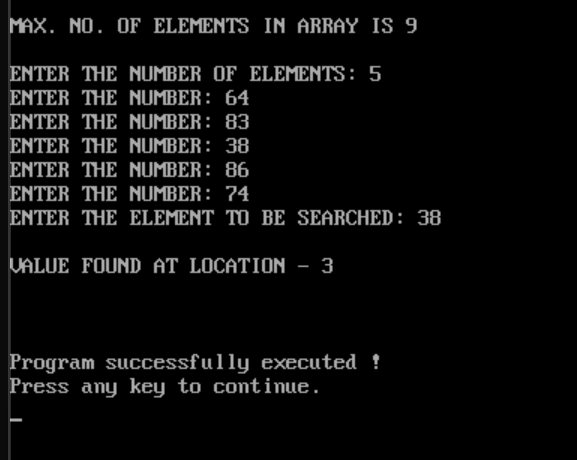
INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

myexit:

MOV DX, OFFSET MESS02 MOV AH, 09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

.EXIT END



Binary-Search :-

.model small ; contain two segment data and code

.stack ; tells the assembler to reserve storage

.386 ;instructions for the 80386 processor

.data ; start of data segment

ARRAY DB 10 DUP(?); Declaring an array with garbage

MESS0 DB 13,10,"ENTER THE NUMBER: $" ; 10 is the ASCII control code for line feed while 13 is the code for carriage return

MESS1 DB 13,10,"ENTER THE NUMBER OF ELEMENTS: $" MESS2 DB 13,10,"ENTER THE ELEMENT TO BE SEARCHED: $" MESS3 DB 13,10,"VALUE FOUND AT LOCATION- $"

MESS4 DB 13,10,"VALUE NOT FOUND!!!$"

ErrMess DB 13,10,"ERROR IN INPUT DIGIT$"

DAT DB ? ;set byte size variable

number dw ? ; set double word size variable

.code ; start of code segment

.startup ; Generates program start-up code

MOV DX,OFFSET MESS1 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

MOV AH,01

INT 21H ; input from user cmp al,39h

jbe abc ; jump if below or equal to

MOV DX,OFFSET ErrMess MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 jmp myexit

abc:

and al,0fh mov ah,0

mov number,ax MOV CX,AX MOV DI,0

MYLOOP:

MOV DX,OFFSET MESS0 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 MOV AH,01

INT 21H ; input from user cmp al,39h

jbe abc2 ; jump if below or equal to MOV DX,OFFSET ErrMess

MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 jmp myexit

abc2:

and al,0fh

MOV ARRAY[DI],AL INC DI

LOOP MYLOOP

MOV DX,OFFSET MESS2 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 MOV AH,01

INT 21H ; input from user cmp al,39h

jbe abc3 ; jump if below or equal to MOV DX,OFFSET ErrMess

MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 jmp myexit

abc3:

and al,0fh MOV DAT,AL

mov ax,ds mov es,ax mov al,dat

CLD ; auto increment mode mov cx,number

INC CX

mov DI, offset ARRAY

repne SCASB ; scan the memory for AL

CMP CX,0

JE NTFOUND

MOV DX,OFFSET MESS3 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9 SUB NUMBER,CX ;FIND ELEMENT LOCATION

ADD NUMBER,30H MOV DX,NUMBER INC DX

MOV AH,02

INT 21H ; Output a character present in DL , as AH value is 2 JMP myexit

NTFOUND:

MOV DX,OFFSET MESS4 MOV AH,09

INT 21H ; Output a string terminated by '$’ stored in DX, as AH =9

myexit:

MOV AH,4CH

INT 21H ; causes the current process to terminate

END

