**Lab 10: File Handling in Assembly**

**OBJECTIVE**

To learn how to deal with files in following ways:

* Opening File
* Reading File
* Detecting Next line in File
* Counting Characters in File
* Writing to File
* Appending File
* Closing File

**File Opening:**

* In EMU8086, create a file “myfile.txt” with some text in directory “C:\emu8086\MyBuild”
* Under data directive, define a string which contains file name and append 0 at end, the last character to write to a file must be a zero.
  + file db “myfile.txt”,0
* Define a buffer in which file contents will be stored after reading, in data segm.
  + buffer db 5000 dup("$")

**Loading File Handler:**

* First step is to Load File handler
  + File handler acts as a pointer to file

mov dx, offset file ; Load address of String “file”

mov al, 0 ; Open file (read-only)

mov ah, 3dh ; Load File Handler and store in ax, function number

int 21h

Reading file:

mov bx,ax ; Move file Handler to bx

mov dx, offset buffer ; Load address of string in which file contents will be stored after reading

mov ah, 3fh ; Interrupt to read file, function number

int 21h ; Read file and store the contents in string whose address is stored in dx

mov ah,9 ; to display the contents of the file

int 21h

**Printing Contents of File:**

* All the contents of file are stored in string “Buffer”.
* Now print this String using int21h function 09h interrupt.

**Detecting Next Line in File:**

* Contents of file is stored in buffer string.
* Next line will be indicated by “0DH” and “0AH”.
* 46h, 41h, 53h, 54h, 0Dh, 0Ah

**Writing to file:**

* First step is to Load File handler
* msg db “Happy$”
  + mov dx, offset file ; Load address of String “file”
  + mov al, 2 ; Open file (read/write)
  + mov ah, 3dh ; Load File Handler and store in ax
  + int 21h

mov cx, 5 ; Number of bytes to write

mov bx,ax ; Move file Handler to bx

mov dx, offset msg ; Load offset of string which is to be written to file

mov ah, 40h ; Write to file, function number

int 21h

**Appending File:**

* Move file pointer to end of file before writing to file

mov cx,0

mov ah, 42h ; Move file pointer

mov al, 02h ; End of File

int 21h

**Closing file:**

mov ah, 3eh ; close the file, function number

int 21h

**Activities:**

* Write assembly program, which opens a file and print its contents on CONSOLE and closes the file.
* Write Assembly program which writes a String to a file. Define String in program.
* Write Assembly program which appends a String to a file.
* Write Assembly program which takes 5 numbers input from user and write to file.

**Check this file handling code:**

.model small

.stack 100h

.data

file2 db "file.txt",0

buffer db 5000 dup("$")

msg db "Hello World","$"

count db 0

.code

mov ax, @data

mov ds, ax

mov dx, offset file2

mov al, 2

mov ah, 3dh ; Open File Handler

int 21h

mov cx,10

mov ah,40h

int 21h

mov bx,ax ; Move file Handler to bxs

mov dx, offset buffer

mov ah, 3fh

int 21h ; Read file and store the contents in address stored in dx

mov dx, offset buffer

mov ah,09

int 21h

mov ah, 4ch

int 21h

end

**Duplicate code for Reading a file.**

.model small

.stack 100h

.data

file db "input.txt",0

buffer db 5000 dup("$")

.code

mov ax,@data

mov ds,ax

;open file in read only mode

;-----------------------------

mov dx,offset file ;load address of strig

mov al,0 ;open in read only mode

mov ah,3dh ;load handler in ax

int 21h

;-----------------------------

mov bx,ax ;load handler in bx

;-----------------------------

mov dx,offset buffer ;load address of string to store

mov ah,3fh ;function to read file

int 21h

;-----------------------------

mov dx,offset buffer ;load address of string

mov ah,09h ;write string on console

int 21h

;-----------------------------

mov ah,4ch

int 21h

end

**Duplicate Code for writing to a file.**

.model small

.stack 100h

.data

file db "output.txt",0

msg db "Fast Nuces$"

.code

mov ax,@data

mov ds,ax

;open file in read and write mode

;-----------------------------

mov dx,offset file ;load address of strig

mov al,2 ;open in read and write mode

mov ah,3dh ;load handler in ax

int 21h

;-----------------------------

mov bx,ax ;load handler in bx

;-----------------------------

mov cx,5 ;how many bytes to write

mov dx,offset buffer ;load address of string to write

mov ah,40h ;function to write file

int 21h

;-----------------------------

;mov dx,offset msg ;load address of string

;mov ah,09h ;write string on console

;int 21h

;-----------------------------

mov ah,4ch

int 21h

end