Name: Prabodh Wankhede

Roll no. 57

Class: SE(B)

Batch – B3

Assignment No. 7

Problem Statement: Write x86 program to sort a list of integers in ascending order. Read input from text file and write the sorted data back to the same text file. Sort using bubble sort.

**MACRO FILE:**

;macro.asm

;macros as per 64 bit conventions

%macro accept 2

mov rax,0 ;read

mov rdi,0 ;stdin/keyboard

mov rsi,%1 ;buf

mov rdx,%2 ;buf\_len

syscall

%endmacro

%macro display 2

mov rax,1 ;print

mov rdi,1 ;stdout/screen

mov rsi,%1 ;msg

mov rdx,%2 ;msg\_len

syscall

%endmacro

%macro fopen 1

mov rax,2 ;2 for open file

mov rdi,%1 ;filename

mov rsi,2 ;mode RW

mov rdx,0777o ;file permission

syscall

%endmacro

%macro fread 3

mov rax,0 ;read

mov rdi,%1 ;filehandle

mov rsi,%2 ;buf

mov rdx,%3 ;buf\_len

syscall

%endmacro

%macro fwrite 3

mov rax,1 ;write/print

mov rdi,%1 ;filehandle

mov rsi,%2 ;buf

mov rdx,%3 ;buf\_len

syscall

%endmacro

%macro fclose 1

mov rax,3 ;close

mov rdi,%1 ;filehandle

syscall

%endmacro

**CODE:**

%include "macro.asm"

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

section .data

nline db 10

nline\_len equ $-nline

ano db 10,10,"ML assignment 7 : Bubble Sort using file operations"

ano\_len equ $-ano

filemsg db 10,"Enter filename of input data :"

filemsg\_len equ $-filemsg

omsg db 10,"Sorting using Bubble Sort operaration successful"

omsg\_len equ $-omsg

errmsg db 10,"ERROER opening/reading/wriring file..",10

errmsg\_len equ $-errmsg

ermsg db 10,"ERROR in writing file...",10

ermsg\_len equ $-ermsg

exitmsg db 10,10,"Exit from program...",10,10

exitmsg\_len equ $-exitmsg

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

section .bss

buf resb 1024

buf\_len equ $-buf

filename resb 50

filehandle resq 1

abuf\_len resq 1

array resb 10

n resq 1

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

section .text

global \_start

\_start:

display ano,ano\_len

;assignment no

display filemsg,filemsg\_len

accept filename,50

dec rax

mov byte[filename+rax],0 ;blank char/null char

fopen filename ;on success return

cmp rax,-1h ;on failure returns -1

je Error

mov [filehandle],rax

fread[filehandle],buf,buf\_len

dec rax ;EOF

mov [abuf\_len],rax

call bsort\_proc

jmp Exit

Error: display errmsg,errmsg\_len

Exit: display exitmsg,exitmsg\_len

mov rax,60

mov rdi,0

syscall

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

bsort\_proc: ;bubble sort

call buf\_array\_proc

mov rax,0

mov rbp,[n]

dec rbp

mov rcx,0

mov rdx,0

mov rsi,0

mov rdi,0

mov rcx,0 ;i=0

oloop: mov rbx,0 ;j=0

mov rsi,array ;a[j]

iloop: mov rdi,rsi ;a[j+1]

inc rdi

mov al,[rsi]

cmp al,[rdi]

jbe next

mov dl,0

mov dl,[rdi] ;swap

mov [rdi],al

mov [rsi],dl

next:inc rsi

inc rbx

cmp rbx,rbp

jb iloop

inc rcx

cmp rcx,rbp

jb oloop

fwrite [filehandle],omsg,omsg\_len

fwrite [filehandle],array,[n]

fclose [filehandle]

display omsg,omsg\_len

display array,[n]

ret

Error1:

display ermsg,ermsg\_len

ret

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

buf\_array\_proc:

mov rcx,0

mov rsi,0

mov rdi,0

mov rcx,[abuf\_len]

mov rsi,buf

mov rdi,array

next\_num:

mov al,[rsi]

mov [rdi],al

inc rsi

inc rsi

inc rdi

inc byte[n]

dec rcx

dec rcx

jnz next\_num

ret

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**OUTPUT:**

9

8

7

4

6

7

Sorting using Bubble Sort operaration successful 467789