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

;Write X86 program to sort the list of integers in ascending/descending order. Read ;the input from the text file and write the sorted data back to the same text file ;using bubble sort)

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

%macro output 2

mov rax,1

mov rdi,1

mov rsi,%1

mov rdx,%2

syscall

%endmacro

%macro input 2

mov rax,0

mov rdi,0

mov rsi,%1

mov rdx,%2

syscall

%endmacro

%macro ffopen 1

mov rax,2

mov rdi,%1

mov rsi,2

mov rdx,0777o

syscall

%endm

%macro ffclose 1

mov rax,3

mov rdi,%1

syscall

%endm

%macro ffread 3

mov rax,0

mov rdi,%1

mov rsi,%2

mov rdx,%3

syscall

%endm

%macro ffwrite 3

mov rax,1

mov rdi,%1

mov rsi,%2

mov rdx,%3

syscall

%endm

;

global fhandle,buf,abuflen ;//File must contain single digit no eg. 6 8 7 1 3 4

global \_start ;//Enter file name with .txt extension

section .data

menu db 10d,'\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'

db 10d,'1.enter filename'

db 10d,'2.ascending order'

db 10d,'3.decending order',0dh,0ah

lenmenu equ $-menu

askinput db 'Enter choice ',0dh,0ah

lenaskinput equ $-askinput

msg1 db 'Enter filename ',0dh,0ah

len1 equ $-msg1

msg2 db 'Error in opening file ',0dh,0ah

len2 equ $-msg2

msg3 db 'Acending: ',0dh,0ah

len3 equ $-msg3

msg4 db 'Decending: ',0dh,0ah

len4 equ $-msg4

section .bss

len resb 2

totalele resb 2

arr resb 20

choice resb 2

value resb 2

opvar resb 2

count1 resb 2

count2 resb 2

count3 resb 2

filename resb 50

buf resb 4096

buflen equ $-buf

fhandle resq 1

abuflen resq 1

temp resb 2

cnt resb 2

section .text

\_start:

mainmenu:

output menu,lenmenu

output askinput,lenaskinput

input choice,2

cmp byte[choice],'1'

je loop1

cmp byte[choice],'2'

je asending

cmp byte[choice],'3'

je decending

jmp exit

loop1:

call openfile

call copyfile ;copy no. in array(arr)

mov bl,byte[abuflen]

mov al,bl

mov bl,02h

div bl

mov byte[totalele],al

mov byte[cnt],al

call display1

jmp mainmenu

asending:

output msg3,len3

mov byte[value],30h

call procedure1

call writefile

jmp mainmenu

decending:

output msg4,len4

mov byte[value],31h

call procedure1

call writefile

jmp mainmenu

exit:

mov rax,60

mov rdi,0

syscall

;------PROCEDURES----------------------

openfile:

output msg1,len1

input filename,50

dec rax

mov byte[filename+rax],0

ffopen filename

cmp rax,-1h

;je errorr

mov [fhandle],rax

ffread [fhandle],buf,buflen

mov [abuflen],rax

ret

copyfile:

mov rsi,arr

mov rbp,buf

mov al,byte[abuflen]

mov byte[cnt],al

up5: mov al,byte[rbp]

mov byte[rsi],al

mov byte[temp],al

inc rbp

inc rbp

inc rsi

dec byte[cnt]

jnz up5

ret

writefile:

mov rax,1

mov rdi,[fhandle]

mov rsi,arr

mov rdx,[abuflen]

syscall

ret

;

display1:

mov rbp,arr

mov bl,byte[cnt]

mov byte[totalele],bl

loop2:

mov al,byte[rbp]

mov byte[opvar],al

output opvar,2

inc rbp

dec byte[totalele]

jnz loop2

ret

;

procedure1:

mov cl,byte[cnt]

mov byte[count1],cl ;loop 1

dec byte[count1]

first:

mov rsi,arr

mov cl,byte[cnt]

mov byte[count2],cl ;loop2

dec byte[count2]

second:

mov al,[rsi]

inc rsi

mov rdi,rsi

dec rsi

mov bl,[rdi]

cmp byte[value],'0'

je asen

jmp dsen

asen:cmp al,bl

ja swap

jmp noswap

dsen:

cmp al,bl

jb swap

jmp noswap

swap:

mov [rdi],al

mov [rsi],bl

noswap:

inc rsi

dec byte[count2]

jnz second

dec byte[count1]

jnz first

call display1

ret

**OUTPUT :**

--------------------------------------------------------

manu@ubuntu:~/mpfinals$ nasm -felf64 ascdsc.nasm

manu@ubuntu:~/mpfinals$ ld -o ascdsc ascdsc.o

manu@ubuntu:~/mpfinals$ ./ascdsc

\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.enter filename

2.ascending order

3.decending order

Enter choice

1

Enter filename

ascdsc.txt

Floating point exception (core dumped)

manu@ubuntu:~/mpfinals$ ./ascdsc

\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.enter filename

2.ascending order

3.decending order

Enter choice

1

Enter filename

numbers.txt

687134

\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.enter filename

2.ascending order

3.decending order

Enter choice

2

Acending:

134678

\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.enter filename

2.ascending order

3.decending order

Enter choice

3

Decending:

876431

\*\*\*\*\*\*\*\*\*MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.enter filename

2.ascending order

3.decending order

Enter choice

4