**Assignment number: 5**

**Subject: MICROPROCESSOR LAB**

Name: ***RIA MITTAL***

Class: ***SECOND YEAR ENGINEERING***

Division: ***B***

Roll no: ***222008***

Batch: ***B1***

**PROBLEM STATEMENT:**

Write X86 ALP to find, a) Number of Blank spaces b) Number of lines c) Occurrence of a particular character. Accept the data from the text file. The text file has to be accessed during Program\_1 execution and write FAR PROCEDURES in Program\_2 for the rest of the processing. Use of PUBLIC and EXTERN directives is mandatory.

**Code:**

extern far\_proc

global fh,char,buff,len

%macro scall 4

mov rax,%1

mov rdi,%2

mov rsi,%3

mov rdx,%4

syscall

%endmacro

section .data

filen db "Enter the filename : "

filen\_l equ $-filen

search db 10,"Enter the character to be searched : "

search\_l equ $-search

err db 10,"Error "

err\_l equ $-err

section .bss

filename resb 50

buff resb 1024

char resb 2

fh resq 1

len resq 1

section .text

global \_start

\_start:

scall 1,1,filen,filen\_l

scall 0,0,filename,50

dec rax

mov byte[filename+rax],0

scall 1,1,search,search\_l

scall 0,0,char,2

scall 2,filename,2,0777o

cmp rax,-1h

jle err\_p

mov [fh],rax

scall 0,[fh],buff,1024

mov [len],rax

call far\_proc

call exit\_p

;----------------------- exit ------------------------------

exit\_p :

mov rax,60

mov rdi,0

syscall

err\_p : scall 1,1,err,err\_l

call exit\_p

**far file:**

extern fh,char,buff,len

global far\_proc

%macro scall 4

mov rax,%1

mov rdi,%2

mov rsi,%3

mov rdx,%4

syscall

%endmacro

section .data

space db 10,"Number of spaces are : "

space\_l equ $-space

no db 10,"Number of lines are : "

no\_l equ $-no

cha db 10,"Number of character occurences : "

cha\_l equ $-cha

nline db 10

nline\_l equ $-nline

r db 10,"ok",10

rl equ $-r

section .bss

scount resq 1

ncount resq 1

ccount resq 1

temp resq 1

ans resb 2

section .txt

global \_main

\_main:

far\_proc:

mov rax,0

mov rbx,0

mov rcx,0

mov rdx,0

mov rdi,0

mov rsi,0

mov bl,[char]

mov rsi,buff

mov rcx,[len]

back : mov al,[rsi]

cmp al,20h

jne line

inc byte[scount]

jmp next1

line : cmp al,0Ah

jne co

inc byte[ncount]

jmp next1

co : cmp al,bl

jne next1

inc byte[ccount]

next1 : inc rsi

dec rcx

jnz back

scall 1,1,space,space\_l

mov rax,[scount]

call disp

;scall 1,1,scount,1

scall 1,1,no,no\_l

mov rax,[ncount]

call disp

;scall 1,1,ncount,1

scall 1,1,cha,cha\_l

mov rax,[ccount]

call disp

;scall 1,1,ccount,1

scall 1,1,nline,nline\_l

scall 3,[fh],0,0

ret

disp :

mov rcx,2

mov rbx,temp

loop : ;mov al,[rdi]

rol al,4

mov dl,al

and dl,0Fh

cmp dl,09h

jle nxt

add dl,07h

nxt : add dl,30h

mov [rbx],dl

inc rbx

;inc rdi

dec rcx

jnz loop

scall 1,1,temp,2

ret

