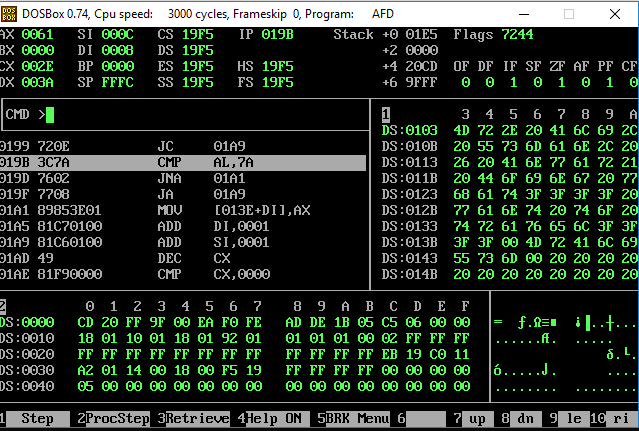
**NAME:**

**Moeez Ali**

**ROLL#: 21L-1798**

**SECTION: D**

**ACTIVITY 1:**



[org 0x0100]

jmp start

msg: db 'Mr. Ali, Moeez, & Amjad! Doing what???? want to travel????',0

msg2:db ' ', 0

remove:

mov dx,cx

mov si,0

mov di,0

l1:

mov bx,0

mov al,[msg+si]

cmp al,'0'

jb no

cmp al,'9'

jbe inp

cmp al,'A'

jb no

cmp al,'Z'

jbe inp

cmp al,'a'

jb no

cmp al,'z'

jbe inp

ja no

inp:

mov [msg2+di],ax

add di,1

no:

add si,1

dec cx

cmp cx,0

jne l1

len:

push ds

pop es

mov di,msg

mov ax,0

mov cx,0xffff

repne scasb

mov ax,0xffff

sub ax,cx

dec ax

ret

prnt:

mov ax,0xb800

mov es,ax

mov ah,0x30

mov si,msg2

mov cx,cx

dec cx

mov di,500

l2:

lodsb

stosw

loop l2

ret

start:

call len

mov cx,ax

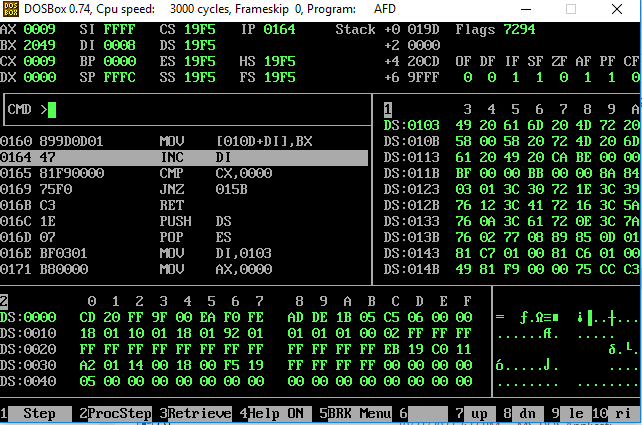
call remove

call prnt

mov ax,0x4c00

int 0x21

**ACTIVITY 2:**



[org 0x0100]

jmp start

msg: db 'I am Mr X',0

msg2:db ' ',0

remove:

mov dx,cx

mov si,0

mov di,0

l1:

mov bx,0

mov al,[msg+si]

cmp al,'0'

jb no

cmp al,'9'

jbe inp

cmp al,'A'

jb no

cmp al,'Z'

jbe inp

cmp al,'a'

jb no

cmp al,'z'

jbe inp

ja no

inp:

mov [msg2+di],ax

add di,1

no:

add si,1

dec cx

cmp cx,0

jne l1

ret

reverse:

mov di,0

mov cx,cx

mov si,cx

dec si

l3:

mov bx,[msg+si]

dec si

mov [msg2+di],bx

inc di

cmp cx,0

jne l3

ret

len:

push ds

pop es

mov di,msg

mov ax,0

mov cx,0xffff

repne scasb

mov ax,0xffff

sub ax,cx

dec ax

ret

prnt:

mov ax,0xb800

mov es,ax

mov ah,0x30

mov si,msg2

mov cx,cx

dec cx

mov di,500

l2:

lodsb

stosw

loop l2

ret

start:

call len

mov cx,ax

;call remove

;call prnt

call reverse

mov ax,0x4c00

int 0x21

**ACTIVITY 3:**

[org 0x0100]

jmp start

msg: db 'aaabbaaa',0

msg2:db ' ',0

msg3:db 'palindrome'

remove:

mov dx,cx

mov si,0

mov di,0

l1:

mov bx,0

mov al,[msg+si]

cmp al,'0'

jb no

cmp al,'9'

jbe inp

cmp al,'A'

jb no

cmp al,'Z'

jbe inp

cmp al,'a'

jb no

cmp al,'z'

jbe inp

ja no

inp:

mov [msg2+di],ax

add di,1

no:

add si,1

dec cx

cmp cx,0

jne l1

ret

reverse:

mov di,0

mov cx,cx

mov si,cx

dec si

l3:

mov bx,[msg+si]

dec si

mov [msg2+di],bx

inc di

cmp cx,0

jne l3

ret

len:

push ds

pop es

mov di,msg

mov ax,0

mov cx,0xffff

repne scasb

mov ax,0xffff

sub ax,cx

dec ax

ret

prnt:

mov ax,0xb800

mov es,ax

mov ah,0x30

mov si,msg3

mov cx,10

dec cx

mov di,500

l2:

lodsb

stosw

loop l2

ret

compare:

mov si,msg

mov di,msg2

mov cx,cx

repe cmpsb

call true

ret

true:

mov ax,0xb800

mov es,ax

mov ah,0x30

mov si,msg3

mov cx,10

mov di,500

l4:

lodsb

stosw

loop l4

ret

start:

call len

mov cx,ax

;call remove

call prnt

call reverse

call compare

mov ax,0x4c00

int 0x21

**ACTIVITY 4&5:**

[org 0x0100]

jmp start

print\_str: push bp

mov bp, sp

push es

push ax

push bx

push cx

push dx

push di

mov ax, 0xb800

mov es, ax

mov ax, [bp+4]

mov bx, 10

mov cx, 0

nextdigit: mov dx, 0

div bx

add dl, 0x30

push dx

inc cx

cmp ax, 0

jnz nextdigit

mov di, [bp+6]

nextpos: pop dx

mov dh, 0x07

mov [es:di], dx

add di, 2

loop nextpos

pop di

pop dx

pop cx

pop bx

pop ax

pop es

pop bp

ret 2

clrscr:

push ax

push es

push cx

push di

mov ax , 0xb800

mov es , ax

xor di , di

mov cx , 2000

mov ax , 0x0720

cld

rep stosw

pop di

pop cx

pop es

pop ax

ret

delay:

push si

mov si , 0xffff

lop1:

dec dx

jnz lop1

pop si

ret

randomposition:

push bp

mov bp, sp

push ax

push cx

push bx

mov al, 80

mul byte [bp+4]

add ax, [bp+4]

shl ax, 1

mov di, ax

pop bx

pop cx

pop ax

pop bp

ret 2

start:

mov ax, 10

push ax

call randomposition ; will update di

call clrscr

mov cx,15

mov bx,0

counter:

mov ax , dx

push ax

mov ax , bx

push ax

call Print

call clrscr

add bx,1

add dx,2

call delay

loop counter

mov ax, 0x4c00

int 21h