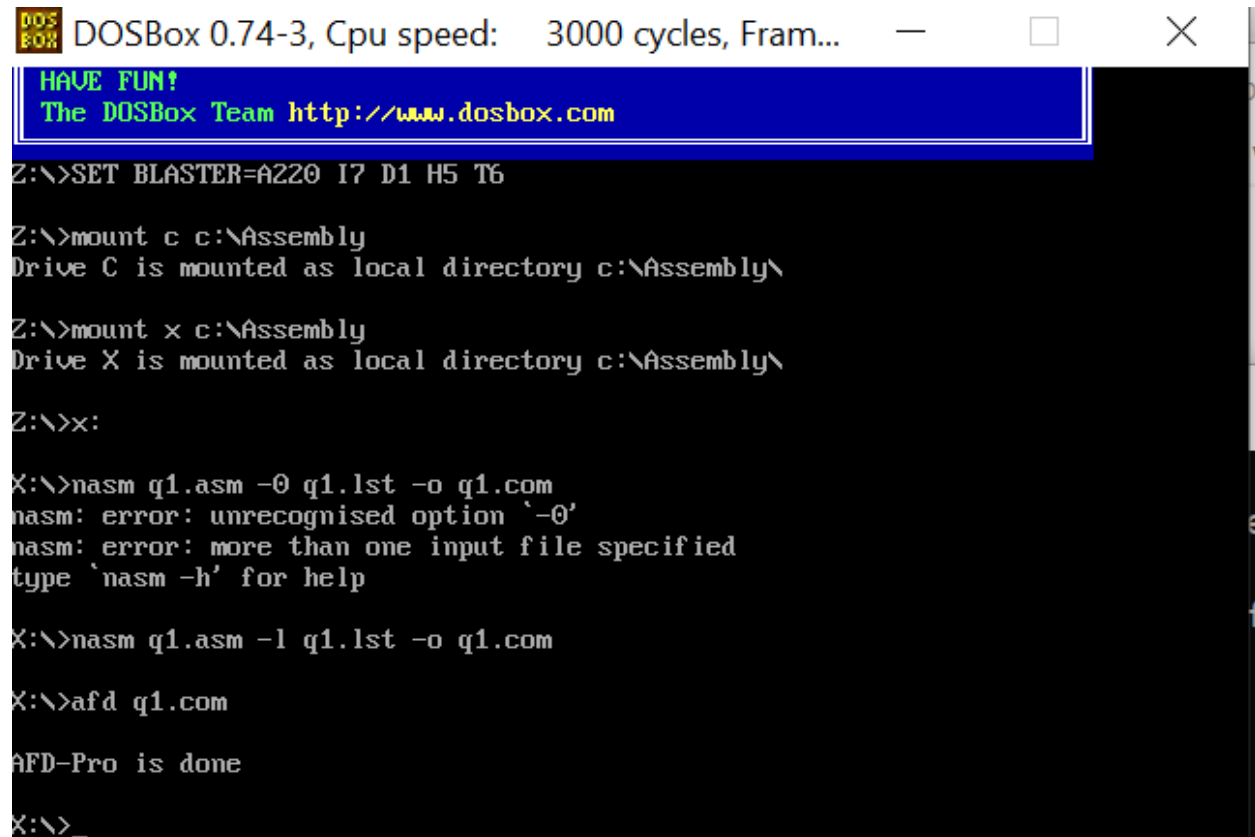


Name : Amber Khurshid

Section: BAI-4A


Roll No: 22P-9295

Compilation :



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...  
HAVE FUN!  
The DOSBox Team http://www.dosbox.com  
Z:\>SET BLASTER=A220 I7 D1 H5 T6  
  
Z:\>mount c c:\Assembly  
Drive C is mounted as local directory c:\Assembly\  
  
Z:\>mount x c:\Assembly  
Drive X is mounted as local directory c:\Assembly\  
  
Z:\>x:  
  
X:\>nasm q1.asm -0 q1.lst -o q1.com  
nasm: error: unrecognised option '-0'  
nasm: error: more than one input file specified  
type 'nasm -h' for help  
  
X:\>nasm q1.asm -l q1.lst -o q1.com  
  
X:\>afd q1.com  
  
AFD-Pro is done  
  
X:\>_
```

Dos box screenshot:


DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...

AX 0000	SI 0000	CS 19F5	IP 0100	Stack +0 0000	Flags 7202
BX 0000	DI 0000	DS 19F5		+2 20CD	
CX 005D	BP 0000	ES 19F5	HS 19F5	+4 9FFF	OF DF IF SF ZF AF PF CF
DX 0000	SP FFFE	SS 19F5	FS 19F5	+6 EA00	0 0 1 0 0 0 0 0

CMD >

0100 E94300	JMP	0146
0103 2D0016	SUB	AX,1600
0106 005589	ADD	[DI-77],DL
0109 E550	IN	AX,[50]
010B 53	PUSH	BX
010C 8B4606	MOV	AX,[BP+06]
010F 8B5E04	MOV	BX,[BP+04]
0112 39D8	CMP	AX,BX

1	0	1	2	3	4	5	6	7
DS:0000	CD	20	FF	9F	00	EA	F0	FE
DS:0008	AD	DE	1B	05	C5	06	00	00
DS:0010	18	01	10	01	18	01	92	01
DS:0018	01	01	01	00	02	FF	FF	FF
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF
DS:0028	FF	FF	FF	FF	EB	19	C0	11
DS:0030	A2	01	14	00	18	00	F5	19
DS:0038	FF	FF	FF	FF	00	00	00	00
DS:0040	05	00	00	00	00	00	00	00
DS:0048	00	00	00	00	00	00	00	00

2	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
DS:0000	CD	20	FF	9F	00	EA	F0	FE	AD	DE	1B	05	C5	06	00	00
DS:0010	18	01	10	01	18	01	92	01	01	01	00	02	FF	FF	FF	FF
DS:0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	EB	19	C0	11
DS:0030	A2	01	14	00	18	00	F5	19	FF	FF	FF	FF	00	00	00	00
DS:0040	05	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

= f.Ω≡ i |..†...
.....ff.
δ.L.
ó.....J.
.....

1 Step 2ProcStep 3Retrieve 4Help ON 5BRK Menu 6 7 up 8 dn 9 le 10 ri

```

Code:

[org 0x0100]

jmp start

num1: dw 45

num2: dw 22

equality:

    push bp
  
```

mov bp, sp

push ax

push bx

mov ax, [bp+6]

mov bx, [bp+4]

cmp ax, bx

jne not_equal

not_equal:

pop bx

pop ax

mov sp, bp

pop bp

ret

less_than_or_equal_to:

push bp

mov bp, sp

push ax

push bx

mov ax, [bp+6]

mov bx, [bp+4]

cmp ax, bx

jg not_less_equal

not_less_equal:

pop bx

pop ax

mov sp, bp

pop bp

ret

greater_than_or_equal_to:

push bp

mov bp, sp

push ax

push bx

mov ax, [bp+6]

mov bx, [bp+4]

cmp ax, bx

jl not_greater_equal

not_greater_equal:

```
pop bx
pop ax
mov sp, bp
pop bp
ret
```

start:

```
mov ax, [num1]
mov bx, [num2]
```

```
push bx
push ax
```

```
call equality
call less_than_or_equal_to
call greater_than_or_equal_to
```

```
mov ax, 0x4c00
int 0x21
```