

NAME: RIMSHA LARAIB	SEAT NUMBER: B21110006107	COURSE CODE: BSCS 403	SECTION: A (MORNING)
--------------------------------	--------------------------------------	----------------------------------	---------------------------------

LAB TASK # 5

1. Submit program using conditional jump performed in today lab.

Code:

The screenshot shows the DOSBox interface with the assembly code for a program named L1.ASM. The code implements a conditional jump based on user input. It uses .model small, .stack 100h, and .data sections. The .code section contains a main procedure that reads a character from the user, compares it with '1', and based on the result, prints one of six messages (msg1 to msg6) using INT 21h. Finally, it exits with AH=4Ch. The assembly code is as follows:

```
;PROGRAM USING CONDITIONAL JUMP_
dosseg
.model small
.stack 100h
.data
msg1 db 'number is less than $'
msg2 db 'number is not less than $'
msg3 db 'number is not greater than $'
msg4 db 'number is greater$'
msg5 db 'number is not equal$'
msg6 db 'number is equal$'
.code
main proc
mov ax, @data
mov ds,ax
mov dl,'?'
mov ah,1
int 21h
cmp al,d1
jl 11
mov dx,offset msg2
mov ah,9
int 21h
je 13
mov dx,offset msg5
mov ah,9
int 21h
jg 12
mov dx,offset msg3
mov ah,9
int 21h
mov ah,4ch
int 21h
13:
mov dx,offset msg6
mov ah,9
int 21h
mov ah,4ch
int 21h
12:
mov dx,offset msg4
mov ah,9
int 21h
mov ah,4ch
int 21h
11:
mov dx,offset msg1
mov ah,9
int 21h
mov ah,4ch
int 21h
main endp
end main
```

F1=Help | Line:54 Col:32

Output:

```
C:\>l1.exe  
7number is not less than number is equal  
C:\>l1.exe  
4number is less than  
C:\>l1.exe  
9number is not less than number is not equal number is greater  
C:\>_
```

2. Submit program using multiply performed in today lab.

Code:

The screenshot shows the DOSBox interface with the title bar "DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT". The menu bar includes File, Edit, Search, View, Options, Help, and the current file name "C:\L2.ASM". The main window displays the assembly code:

```
;PROGRAM TO MULTIPLY TWO NUMBERS  
.dosseg  
.model small  
.stack 100h  
.data  
.code  
main proc  
mov al,9  
mov bl,2  
mul bl  
aam  
mov ch,ah  
mov cl,al  
mov dl,ch  
add dl,48  
mov ah,2  
int 21h  
mov dl,cl  
add dl,48  
mov ah,2  
int 21h  
mov ah,4ch  
  
int 21h  
main endp  
end main
```

The status bar at the bottom shows "F1=Help" and "Line:27 Col:1".

Output:

```
C:\>l2.exe  
18  
C:\>_
```

3. Your code using Conditional jump & multiply as discussed in class.

Code:

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search Options Help C:\A3.ASM

```
;program which prints the multiple of 2
dosseg
.model small
.stack 100h
.data
m1 db 'loop ends $'
.code
main proc
mov ax,@data
mov ds,ax
mov cx,9
mov bl,1
l1:
mov al,2
mul bl
mov dx,ax
add dx,48
cmp dx,'9'
jg l2
mov ah,2
int 21h
inc bl
loop l1
mov ah,4ch
int 21h
l2:
aam
mov ch,ah
mov cl,al
mov dl,ch
add dl,48
mov ah,2
int 21h
mov dl,cl
add dl,48
mov ah,2
int 21h
cmp bl,'9'
je l3
inc bl
jmp l1
mov ah,4ch
int 21h
l3:
```

```
cmp bl,'9'  
je 13  
inc bl  
jmp 11  
mov ah,4ch  
int 21h  
13:  
mov dx,10  
mov ah,2  
int 21h  
mov dx,13  
mov ah,2  
int 21h  
mov dx,offset m1  
mov ah,9  
int 21h  
mov ah,4ch  
int 21h  
  
main endp  
end main
```

F1=Help

Line:59 Col:25

Output:

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX  
C:\>edit a3.asm  
C:\>edit a3.asm  
C:\>masm a3.asm;  
Microsoft (R) Macro Assembler Version 5.00  
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.  
  
51680 + 464864 Bytes symbol space free  
  
0 Warning Errors  
0 Severe Errors  
C:\>link a3.obj;  
Microsoft (R) Overlay Linker Version 3.60  
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.  
  
C:\>a3.exe  
24681012141618202224262830323436384042444648505254565860626466687072747678808284  
86889092949698:0:2:4:6:8:0:2:4  
loop ends  
C:\>
```

4. Program to print an Array using Loop.

Code:

The screenshot shows the DOSBox interface with the title bar "DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT" and the file path "C:\L3.ASM". The menu bar includes File, Edit, Search, View, Options, and Help. The assembly code is displayed in the main window:

```
;PROGRAM TO PRINT AN ARRAY
dosseg
.model small
.stack 100h
.data
a db 'element of an array $'
ar1 db ?,?,?
.code
main proc
mov ax,@data
mov ds,ax
mov si,offset ar1
mov cx,3
l1:
mov dx,offset a
mov ah,9
int 21h
mov ah,1
int 21h
mov dx,[si]
mov ah,2
int 21h
mov dx,10
mov ah,2
int 21h
mov dx,13
mov ah,2
int 21h
inc si
loop l1
mov ah,4ch
int 21h
main endp
end main
```

The status bar at the bottom shows "F1=Help" and "Line:36 Col:1".

Output:

The screenshot shows the DOSBox interface with the command "C:\>13.exe" entered. The output window displays the following text:

```
C:\>13.exe
element of an array 1
element of an array 2
element of an array 3

C:\>13.exe
element of an array r
element of an array i
element of an array m

C:\>_
```

5. Program to input string and print it.

Code:

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search View Options Help C:\S1.ASM

```
:PROGRAM TO INPUT STRING AND PRINT IT
dosseg
.model small
.stack 100h
.data
arr db 40 dup ('$')
.code
main proc
mov ax,@data
mov ds,ax
mov si,offset arr
l1:
mov ah,1
int 21h
cmp al,13
je l2
mov [si],al
inc si
jmp l1
l2:
mov dx,offset arr
mov ah,9
int 21h

mov ah,4ch
int 21h
main endp
end main
```

F1=Help | Line:30 Col:38

Output:

```
C:\>s1.exe
Rimsha Laraib
Rimsha Laraib
C:\>
```

6. Your code using Array & String as discussed in class.

Code:

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT

File Edit Search View Options Help C:\NAZ.ASM

```
dosseg
.model small
.stack 100h
.data
arr1 db 100 dup('$')
msg1 db 'input is:$'
msg2 db 'output is:$'
.code
main proc
mov ax,@data
mov ds,ax
mov si,offset arr1
mov cx,4
l1:
;mov dx,offset msg1
;mov ah,9
;int 21h
;mov ah,1
;int 21h
;cmp al,13
;je pend1
;mov [si],al
inc si
jmp l1
pend1:
mov dx,offset msg2
mov ah,9
int 21h
mov dx,offset arr1
mov ah,9
int 21h
mov dx,10
mov ah,2
int 21h
mov dx,13
mov ah,2
int 21h
loop l1
mov ah,4ch
int 21h
main endp
end main
```

F1=Help | Line:22 Col:1

Output:

```
C:\>a2.exe
RIMSHA LARAIB
output is:RIMSHA LARAIB
D/O M.SHARF-UZ-ZAMAN
output is:RIMSHA LARAIB D/O M.SHARF-UZ-ZAMAN
PLACE TO LIVE IS KARACHI
output is:RIMSHA LARAIB D/O M.SHARF-UZ-ZAMAN PLACE TO LIVE IS KARACHI
output is:RIMSHA LARAIB D/O M.SHARF-UZ-ZAMAN PLACE TO LIVE IS KARACHI
C:\>
```