



## **AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY**

### **Department of Computer Science and Engineering**

**Course No** : CSE 2214  
**Course Title** : Assembly Language Programming Sessional  
**Assignment no** : 09  
  
**Date of Performance** : 26.08.2020  
**Date of Submission** : 01.09.2020  
**Submitted To** : Ms.Tahsin Aziz & Md.Siam Ansary

#### **Submitted By:**

**Group** : A1  
**Name** : Nusrat Jahan  
**Id** : 18.01.04.020  
**Section** : A

Question no: 01

Question: Write a program that lets the user enter time in seconds, up to 65535 and outputs the time as hours, minutes, and seconds.

Solution:

```
.MODEL SMALL
.STACK 100H
.DATA
    M1 DB 0AH,0DH,'Enter the time in seconds up to 65535 = $ '
    M2 DB 0AH,0DH,'The time in hh:mm:ss format is = $'
.CODE
    MAIN PROC
        MOV AX, @DATA
        MOV DS, AX
        LEA DX, PROMPT_1
        MOV AH, 9
        INT 21H
    CALL INDEC
        MOV DX,0
        MOV BX,3600
        DIV BX
        PUSH AX
        PUSH DX
        MOV AH,9
```

```
        LEA DX,M2
        INT 21H
        POP DX
        POP AX
CALL OUTDEC
        PUSH DX
        MOV AH,2
        MOV DL,':'
        INT 21H
        POP AX
        MOV BX,60
        MOV DX,0
        DIV BX
CALL OUTDEC
        PUSH DX
        MOV AH,2
        MOV DL,':'
        INT 21H
        POP AX
CALL OUTDEC
        MOV AH, 4CH
        INT 21H
MAIN ENDP

INCLUDE D:\ASSEMBLY\TEST_9A\OUTDEC_9(a).ASM
INCLUDE D:\ASSEMBLY\TEST_9A\INDEC_9(a).ASM
END MAIN
```

## INDEC PROC

PUSH BX

PUSH CX

PUSH DX

BEGIN:

MOV AH,2

MOV DL,20H

INT 21H

XOR BX,BX

XOR CX,CX

MOV AH,1

INT 21H

CMP AL,'-'

JE MINUS

CMP AL,'+'

JE PLUS

JMP REPEAT2

MINUS:

MOV CX,1

PLUS:

NT 21H

REPEAT2:

CMP AL,'0'

JNGE NOT\_DIGIT

CMP AL,'9'

JNLE NOT\_DIGIT

AND AX,000FH

PUSH AX

MOV AX,10

```
MUL BX
POP BX
ADD BX,AX
MOV AH,1
INT 21H
CMP AL,0DH
JNE REPEAT2
MOV AX,BX
OR CX,CX
JE EXIT
NEG AX
EXIT:
POP DX
POP CX
POP BX
RET
NOT_DIGIT:
MOV AH,2
MOV DL,0DH
INT 21H
MOV DL,0AH
INT 21H
JMP BEGIN
INDEC ENDP
```

```
OUTDEC PROC
```

```
PUSH BX
PUSH CX
PUSH DX
```

```
CMP AX, 0
JGE @START
PUSH AX
MOV AH, 2
MOV DL, "-"
INT 21H
POP AX
NEG AX
@START:
XOR CX, CX
MOV BX, 10
@OUTPUT:
XOR DX, DX
DIV BX
PUSH DX
INC CX
OR AX, AX
JNE @OUTPUT
MOV AH, 2
@DISPLAY:
POP DX
OR DL, 30H
INT 21H
LOOP @DISPLAY
POP DX
POP CX
POP BX
RET
OUTDEC ENDP
```

Question no: 02

Question: Write a program to find the greatest common divisor (GCD) of two integers M and N, according to the following algorithm:

- Divide M by N, getting quotient Q and remainder R.
- If  $R = 0$  then stop. N is the GCD of M and N.
- If  $R \neq 0$  replace M by N, N by R, and repeat step 1

Solution:

```
.MODEL SMALL
.STACK 100H
.DATA
    M1 DB 'Enter the value of M = $'
    M2 DB 0DH,0AH,'Enter the value of N = $'
    M3 DB 0DH,0AH,'The GCD of M and N is = $'
.CODE
    MAIN PROC
        MOV AX, @DATA
        MOV DS, AX
        LEA DX, M1
        MOV AH, 9
        INT 21H
        CALL INDEC
        PUSH AX
        LEA DX, M2
        MOV AH, 9
        INT 21H
```

```
CALL INDEC

    PUSH AX

    POP BX

    POP AX

L1: MOV DX,0

    DIV BX

    CMP DX,0

    JE GCD_FOUND

    MOV AX,BX

    MOV BX,DX

    JMP L1

GCD_FOUND:

    LEA DX, M3

    MOV AH, 9

    INT 21H

    MOV AX, BX

CALL OUTDEC

    MOV AH, 4CH

    INT 21H

MAIN ENDP

INCLUDE D:\ASSEMBLY\TEST_9B\OUTDEC_9(b).ASM

INCLUDE D:\ASSEMBLY\TEST_9B\INDEC_9(b).ASM

END MAIN
```

```
INDEC PROC

    PUSH BX

    PUSH CX

    PUSH DX
```



```
BEGIN:
MOV AH,2
MOV DL,20H
INT 21H
XOR BX,BX
XOR CX,CX
MOV AH,1
INT 21H
CMP AL,'-'
JE MINUS
CMP AL,'+'
JE PLUS
JMP REPEAT2
MINUS:
MOV CX,1
PLUS:
INT 21H
REPEAT2:
CMP AL,'0'
JNGE NOT_DIGIT
CMP AL,'9'
JNLE NOT_DIGIT
AND AX,000FH
PUSH AX
MOV AX,10
MUL BX
POP BX
ADD BX,AX
MOV AH,1
```

```
    INT 21H
    CMP AL,0DH
    JNE REPEAT2
    MOV AX,BX
    OR CX,CX
    JE EXIT
    NEG AX
EXIT:
    POP DX
    POP CX
    POP BX
    RET
NOT_DIGIT:
    MOV AH,2
    MOV DL,0DH
    INT 21H
    MOV DL,0AH
    INT 21H
    JMP BEGIN
    RET
INDEC ENDP
```

```
OUTDEC PROC
```

```
    PUSH AX
    PUSH BX
    PUSH CX
    PUSH DX
    OR AX,AX
```

```
JGE END_IF1
PUSH AX
MOV DL,'-'
MOV AH,2
INT 21H
POP AX
NEG AX
END_IF1:
XOR CX,CX
MOV BX,10D
REPEAT1:
XOR DX,DX
DIV BX
PUSH DX
INC CX
OR AX,AX
JNE REPEAT1
MOV AH,2
PRINT_LOOP:
POP DX
OR DL,30H
INT 21H
LOOP PRINT_LOOP
POP DX
POP CX
POP BX
POP AX
RET
OUTDEC ENDP
```