# Ahsanullah University of Science & Technology

## Department of Computer Science & Engineering

**Course No** : CSE2214
**Course Title** : Assembly Language Programming Sessional
**Assignment No** : 08

**Date of Performance** : 20.07.2023
**Date of Submission** : 27.07.2023

**Submitted To** : Ms. Tahsin Aziz & Mr. Md. Zahid Hossain


**Submitted By-**
**Group** : $A_2$
**Name** : Afia Fahmida
**Id** : 20210104032
**Section** : A

Answer 1:

```
.MODEL SMALL
.STACK 100H
.DATA
   MSG1 DB 'Enter a string: $'
   MSG2 DB 13,10,'Reverse of the String: $'

.CODE
  MAIN PROC
      MOV AX, @DATA
      MOV DS, AX.

      MOV AH, 9
      LEA DX, MSG1
      INT 21H

      WHILE:
      CMP AL, 0DH
      JE END-WHILE

      PUSH AX
      INC CX
      INT 21H
      JMP WHILE

      END WHILE:
      MOV AH, 9
      LEA DX, MSG2
      INT 21H.
      MOV AH,2
      MOV DL, 0DH
      INT 21H
      MOV DL, 0AH
      INT 21H
      JCXZ EXIT

      TOP:
      POP DX
      INT 21H
      LOOP TOP

      EXIT:
      MOV AH, 4CH
      INT 21H
      MAIN ENDP
END MAIN.
```

Output:

```
Enter a string: CSE is fun
Reverse of the string:
nuf si ESC.
```

Answer : 2

```
.MODEL SMALL
.STACK 100H

.DATA
  ENTER   DB  0DH,0AH,'Enter an Algebraic Expression: $'
  CORRECT DB  0DH,0AH,'expression is Correct $'
  LEFT    DB  0DH,0AH,'too many left brackets $'
  RIGHT   DB  0DH,0AH,'too many right brackets $'
  AGAIN   DB  0DH,0AH,'Mismatch!! $'
  Y       DB  0DH,0AH,'Type Y if you want to continue : $'

.CODE
 MAIN PROC
    MOV AX,@DATA
    MOV DS,AX

    START:
    MOV AH,9
    LEA DX,ENTER
    INT 21H

    XOR CX,CX
    MOV AH,1

   INPUT:
    INT 21H
    CMP AL,0DH
    JE END_INPUT
    CMP AL,"["
    JE BRACKET
    CMP AL,"{"
    JE BRACKET
    CMP AL,"("
    JE BRACKET
    CMP AL,")"
    JE ROUND
    CMP AL,"}"
    JE CURLY
    CMP AL,"]"
    JE SQUARE

    JMP INPUT
```

```
BRACKET:
PUSH AX
INC CX
JMP INPUT

ROUND:
POP DX
DEC CX
CMP CX,0
JL R_BRACKETS
CMP DL, ")"
JNE MISMATCH
JMP INPUT

CURLY:
POP DX
DEC CX
CMP CX,0
JL R_BRACKETS
CMP DL, "}"
JNE MISMATCH
JMP INPUT

SQUARE:
POP DX
DEC CX
CMP CX,0
JL R_BRACKETS

CMP DL, "]"
JNE MISMATCH
JMP INPUT

END INPUT:
CMP CX,0
JNE L_BRACKETS

MOV ,AH,9
LEA DX, CORRECT
INT 21H
```

```
LEA DX, Y
INT 21H

MOV AH, 1
INT 21H

CMP AL, "Y"
JNE EXIT
JMP START

MISMATCH:
MOV AH, 9
LEA DX, AGAIN
INT 21H
JMP START

L_BRACKETS:
MOV AH, 9
LEA DX, LEFT
INT 21H
JMP START

R_BRACKETS:
MOV AH, 9
LEA DX, RIGHT
INT 21H

JMP START

EXIT:
MOV AH, 4CH
INT 21H
MAIN ENDP
END MAIN.
```

Output:

```
Enter an Algebraic Expression:
3x^2 (((7+6)
too many left Brackets
Enter an Algebraic Expression:
3+(2+8}
Mismatch!!
Enter an Algebraic Expression:
3+(2+8)
expression is Correct
Type Y if you want to Continue:-
```