

### **Experiment 06 : String Operations Part-II**

**Learning Objective:** Student should be able to apply string operations (i) Accept, (ii) Display, (iii) Reverse and (iv) Palindrome in ALP.

**Tools:** TASM/MASM

**Theory:**

**String Instruction:**

#### **1.REP (repeat):**

- This is an instruction prefix, which can be used in string instructions.
- It can be used along with any of the above string instructions only.
- It causes the instruction to be repeated CX number of times.
- After each execution, the SI and DI register are incremented/ decremented based on the DF (Direction flag) in the flag register and CX is decremented. i.e. DF = 1; SI,DI decrements.
- Thus it is important that before we use the REP instruction prefix the following steps must be carried out:
- CX must be initialized to the count value.
- If auto decrementing is required, DF must be cleared using CLD instruction else set using STD instruction.
- Eg:   MOV CX, 0023H  
          ▪   CLD  
          REP MOVSB  
          .  
          .  
          .

The above section of a program will cause the following string operation.

MOVSB   ;ES:[DI] ← DS:[SI], CX ← CX – 1, SI ← SI + 1, DI ← DI + 1

To be executed 23H times (as CX = 23H) in auto incrementing mode (as DF is cleared).

#### **2.REPZ/ REPE (repeat on Zero/ Equal):**

- It is a conditional repeat instruction prefix.
- It behaves the same as a REP instruction provided the Zero Flag is set (i.e. ZF = 1).

### **3.REPNZ/ REPNE (repeat on No Zero/ Not Equal):**

- It is a conditional repeat instruction prefix.
- It behave the same as a REP instruction provided the Zero Flag is reset (i.e. ZF = 0).

### **Explanation :**

Using Macro display the Menu for entering string, calculate length, reverse, palindrome and exit. Accept the choice from user using INT 21H function 01H.

If choice = 1, call procedure for accepting string. Using interrupt INT 21H, function 0AH accept the string and end procedure. Return back to display Menu.

If choice = 2, call procedure to display string and return back to display Menu. (Repeat the process to enter second string)

If choice = 3, call procedure to reverse the string. Display the reversed string and return back to display Menu.

If choice = 4, call procedure to find palindrome of string. If string is palindrome, display palindrome otherwise display String is not palindrome.

If choice = 5, terminate the program. If any other key is pressed display invalid choice.

### **5. Procedure/Algorithm:**

#### **Algorithm:**

Step I: Initialize the data and stack memory.

Step II: Using Macro display Menu.

1. Accept 2. Display 3. Reverse 4. Palindrome 5. Exit.

Step III: Accept choice from user using INT 21H, function 01H.

Step IV: IS choice = 1 jump to step XI else goto step V.

Step V: IS choice = 2 jump to step XIV else goto step VI.

Step VI: IS choice = 3 jump to step XVII else goto step VII.

Step VII: IS choice = 4 jump to step XX else goto step VIII.

Step VIII: IS choice = 5 jump to step XXIII else goto step IX.

Step IX: Display Wrong choice.

Step X: Jump to step II.

Step XI: Call procedure accept.

Step XII: Accept string using INT 21H, function 0AH.

Step XIII: Return to main program and goto step II.

Step XIV: Call procedure display.

Step XV: display string using INT 21H, function 02H.

Step XVI: Return back to main program and jump to step II.

Step XVII: Call procedure Reverse.

Step XVIII: Reverse the string and display.

Step XIX: Return back to main program and jump to step II.

Step XX: Call procedure Palindrome.

Step XXI: Check if string is palindrome. If yes display string is palindrome otherwise are string is not palindrome.

Step XXII: Return back to main program and jump to step II.

Step XXIII: Terminate the program and stop.

**Application:** Use of string operations in the Assembly Language programming to write modular program.

## Design:

## CODE:

```
EXP6.ASM
1 ;STRING OPERATIONS
2 ;EXP6: OPERATIONS: REVERSE AND PALANDROME
3
4 .MODEL SMALL
5 .STACK
6 .DATA
7
8 M1 DB 10,13,"STRING ACCEPT: $"
9 M2 DB 10,13,"STRING LENGTH : $"
10 M3 DB 10,13,"STRING DISPLAY: $"
11 M4 DB 10,13,"STRING REVERSE: $"
12 M5 DB 10,13,"STRING PALINDAROME: $"
13 M6 DB 10,13,"STRING NOT PALINDAROME: $"
14
15 STR0 DB 50,?,50 DUP(?)
16 STR1 DB 50,?,50 DUP(?)
17 LEN DB ?
18
19 .CODE
20
21 DISP MACRO XX
22     MOV AH,09
23     LEA DX,XX
24     INT 21H
25 ENDM
26
27 .STARTUP
28
29 DISP M1 ;ENTER STRING
30
31 MOV AH,0AH
32 LEA DX,STR0
33 INT 21H
34
35 DISP M2 ;LENGTH STRING
36 LEA SI,STR0+1
37
```

```
EXP6.ASM
37
38 MOV CL,[SI] ;CL LENGTH STRING
39 MOV LEN,CL
40 MOV DL,CL
41 ADD DL,30H
42 MOV AH,02
43 INT 21H
44
45
46 DISP M3 ;DISPLAY STR3
47
48 MOV CL,LEN
49 LEA SI,STR0+2
50
51 BACK:
52 MOV DL,[SI]
53 MOV AH,02
54 INT 21H
55 INC SI
56 DEC CL
57 JNZ BACK
58
59 MOV CL,LEN
60 MOV CH,CL
61
62 LEA SI,STR0+2
63 LEA DI,STR1+2
64 DISP M4
65
66 BACK1:
67 INC SI
68 DEC CL
69 JNZ BACK1
70
71 DEC SI
72
73 BACK2:
```

```
EXP6.ASM
73 BACK2:
74 MOV DL,[SI]
75 MOV [DI],DL
76 MOV AH,02
77 INT 21H
78 DEC SI
79 INC DI
80 DEC CH
81 JNZ BACK2
82
83 ;PALINDROME
84
85 MOV CL,LEN
86 LEA SI,STR0+2
87 LEA DI,STR1+2
88
89 BACK4:
90 MOV DL,[SI]
91 CMP DL,[DI]
92 JNZ AA
93 INC SI
94 INC DI
95 DEC CL
96 JNZ BACK4
97 DISP M5
98
99 AA:
100 DISP M6
101
102 LAST:
103
104
105 END
```

OUTPUT:

```
DOS BOX DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...
C:\>TASM EXP6.ASM
Turbo Assembler Version 2.51 Copyright (c) 1988, 1991 Borland International

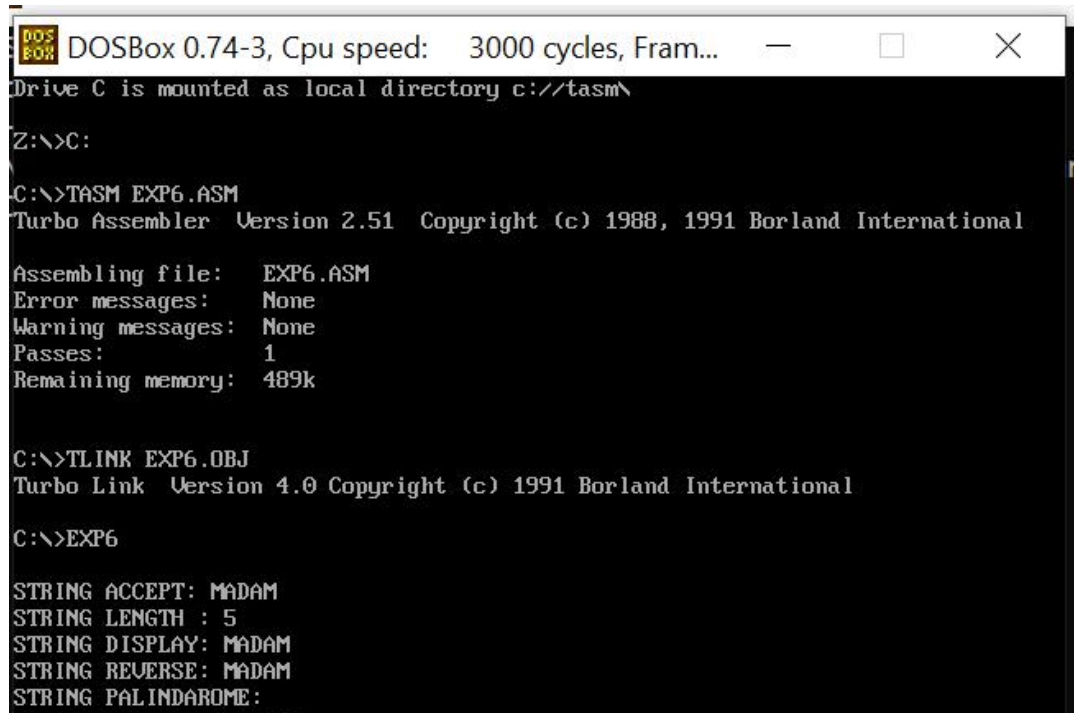
Assembling file: EXP6.ASM
Error messages: None
Warning messages: None
Passes: 1
Remaining memory: 489k

C:\>TLINK EXP6.ASM
Turbo Link Version 4.0 Copyright (c) 1991 Borland International
Fatal: Bad object file exp6.asm

C:\>TLINK EXP6.OBJ
Turbo Link Version 4.0 Copyright (c) 1991 Borland International

C:\>EXP6.EXE

STRING ACCEPT: COMP
STRING LENGTH : 4
STRING DISPLAY: COMP
STRING REVERSE: PMOC
STRING NOT PALINDAROME:
```



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Fram...
Drive C is mounted as local directory c://tasm\
Z:\>C:
C:\>TASM EXP6.ASM
Turbo Assembler Version 2.51 Copyright (c) 1988, 1991 Borland International
Assembling file: EXP6.ASM
Error messages: None
Warning messages: None
Passes: 1
Remaining memory: 489k

C:\>TLINK EXP6.OBJ
Turbo Link Version 4.0 Copyright (c) 1991 Borland International

C:\>EXP6

STRING ACCEPT: MADAM
STRING LENGTH : 5
STRING DISPLAY: MADAM
STRING REVERSE: MADAM
STRING PALINDAROME:
```

## **Result and Discussion:**

**Learning Outcomes:** The student should have the ability to

- LO1: List the string instructions
- LO2: Describe the string addressing mode.
- LO3: Use of string instructions in the program to perform different string operations.

**Course Outcomes:** Upon completion of the course students will be able to make use of instructions of 8086 to build assembly and Mixed language programs.

## **Conclusion:**

Thus we've successfully implemented String Operations Part-II.

**Viva Questions:**

1. Which function is used to accept a string?
2. Explain the procedure to reverse a string
3. Explain the procedure to check whether a string is palindrome or not.

For Faculty Use

---

<b>Correction Parameters</b>	<b>Formative Assessment [40%]</b>	<b>Timely completion of Practical [ 40%]</b>	<b>Attendance / Learning Attitude [20%]</b>	
<b>Marks Obtained</b>				