**Assignment No. 3**

# Aim:-

Write ALP to perform following operation on string:

i. Find and display length

ii. Display reverse

iii. Check whether string is palindrome or not.

# OBJECTIVES:-

1. To study functions of INT 21h to accept string.
2. To study string instructions with their specific significance.
3. To study use of NEAR procedures.

**THEORY:-**

**ASSEMBLER DIRECTIVES USED:**-

**PROC and ENDP directives:**

They are used to define procedures in assembly language programs. They mark the beginning

and end of the procedure.

Syntax:



ENDP is always preceded by either RET or IRET instruction.

**DECLARATION FOR STRING**  :-

mov ah, 0Ah ;Function value to accept string

leadx,str ;load effective address of str into dx register

int 21h

String declaration for 0ah (function value), int 21h

str db 25,?,25 dup('$')

In this declaration first 25 indicated the maximum number of characters user can accept.

"?" mark will indicate the length of actual accepted characters.

From third byte actual string will start. Initially all locations will contain '$"

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 25 | ? | '$' | '$' | '$' | '$' | '$' | '$' | '$' |
| Count | Actual string length | Character starts from here | | | | | | |

**LIST OF MACROS USED** :-

Mess Macro Msg

It is used to display a message.

Input : msg (dummy parameter)

Output : It displays the message on the screen.

Accept Macro var

It is used to accept string.

Input : Accepted string

Output : String will be stored at offset location var.

**LIST OF PROCEDURES USED :-**

1. DispLength proc → It is used to display the length of the string.

Input : Actual length of the string.

Output : It displays the length of the string.

1. Reverse procedure → It is used to reverse a string.

Input : the string which is to be reversed.

Output : Reversed string.

1. Palindromeprocedure → It is used to check whether string is palindrome or not.

Input : String which is to be checked for palindrome.

Output : It states whether the given string is palindrome or not.

**ALGORITHM FOR MAIN PROGRAM :**

1. Physical initialization of data segment.
2. Using Macro display menu. 1. Accept 2. Length 3. Reverse 4. palindrome 5. Exit.
3. Accept choice from the user.
4. Is choice = 1, accept the string. Set flag = 01.
5. Is choice = 2 jump to step 10 else go to step 6.
6. Is choice = 3 jump to step 11 else go to step 7.
7. Is choice = 4, jump to step 12 else got step 8.
8. Is choice = 5, jump to step 13 else goto step 9.
9. Display wrong choice.
10. If flag = 01, call procedure disp else goto step 2.
11. If flag = 01, call procedure reverse else goto step 2.
12. If flag = 01, call procedure palindrome else goto step 2.
13. Terminate the program and stop.

**ALGORITHM FOR DISPLAY LENGTH PROCEDURE :**

1. Store the actual length of the string in some register.
2. Get the MSB of length and convert it to the ASCII value..
3. Display the MSB of the length.
4. Get the LSB of length and convert it to the ASCII value.
5. Display LSB of the length.
6. Return. .

**ALGORITHM FOR REVERSE PROCEDURE :**

1. Initialize memory pointer to point to the last character of the string & a register to the length of the string.
2. Display the character pointed by memory pointer.
3. Decrement memory pointer.
4. Decrement length counter.
5. Jump to Step 2 if not zero.
6. Return

**ALGORITHM FOR PALINDROME :**

1. Initialize 2 memory pointers, one to point to the first character and other to point to the last character of the string.
2. Initialize one register to actual length and move it to AX. Initialize CL to actual length.
3. Use div instruction (divide by 2) and store the quotient as counter.
4. Compare the characters pointed by memory pointers.
5. If not equal jump to step 10 else goto step 6.
6. Decrement the counter and memory pointer(last)
7. Increment first memory pointer.
8. Jump to step 4 if not zero(counter)
9. Display string is palindrome
10. Display string is not a palindrome.
11. Return.

**CONCLUSION:**