

Pune Institute of Computer Technology, Pune
Department of Computer Engineering

Sub :- MPL

Date:- 26/05/2020

Name: - Shrikrushna Santosh Zirape

Roll No: - 21286

Assignment No 2

Problem Statement :-

Write an X86/64 ALP to accept a string and to display its length.

Sw/Hw requirements :-

Intel i5 8th generation 64 bit processor

OS - Ubuntu 16.0 LTS

Editor : - VS- Code , Gedit

Assembler: Nasm

Debugger : gdb

Theory :-

In this program we have to accept the string from the user. The string is accepted character by character and the count for the numbers of the characters accepted is maintained. This count is the length of the string. Also to detect the end of string we have to scan for enter i.e. ASCII 10.

System calls :

Write System Call :

Mov rax, 1

Mov rdi , 1

```
Mov rsi, msg
Mov rdx, 0BH
Syscall
```

Read System Call :

```
Mov rax, 1
Mov rdi , 1
Mov rsi, msg
Mov rdx, 0BH
Syscall
```

Exit System call

```
Mov rax, 60
Mov rsi, 00
syscall
```

Instructions Used :-

1. CMP -

Description: This is going to subtract source and destination operand and result is reflected in the following flag registers.

Flags: CF, ZF

e.g. cmp eax, ecx

2. add:

Description: This instruction adds a number from source to number from destination and puts the result to specified destination.

destination=destination+source

Flags: CF, ZF, OF, PF

e.g. add eax, ebx

3. jnz:

Description: This instruction is used to jump to next instruction in the program

when the zero flag is not equal to 0.

Flags: Only the ZF is affected.

Example: JNZ L1

4. DEC:

Description: This instruction subtracts 1 from the destination word, double word

or byte.

Flags: SF, ZF, OF, PF and AF are affected.

Example: DEC AL

5. INC:

Description: This instruction adds 1 to the destination operand.

Flags: SF, PF, OF, ZF, AF are affected.

Example: INC CX

Algorithm :-

1. Start
2. Declare two variable temp and arr and word
3. Print msg "Enter the string"
4. Accept the string
5. Store length in another variable
6. print the string
7. Hex to Ascii conversion

- a. Point edi to buffer
 - b. Rotate number by left four bits
 - c. Move lower byte in dl
 - d. Mask upper digit of byte in dl
 - e. Add 30 h to calculate ascii code
 - f. Compare with 39 h
 - g. If less skip adding 7 more
 - h. Else add 7
 - i. Store ascii code in buffer
 - j. Decrement the counter of digit to display
 - k. If not zero jump to repeat
8. Print the value of buffer array(length of string)
 9. end

Conclusion :-

Hence we found out the length of string and displayed it using assembly language

Program : -

```
%macro mcr 3
mov rax, %1 ;first argument
mov rdi, 01
mov rsi, %2 ;second argument
mov rdx, %3 ;third argument
syscall
%endmacro mcr

section .data
    msg1 db "enter string - "
    msg2 db "The string is - "
```

```
msg3 db "The length is - "  
newline db "",10
```

```
section .bss  
    wrd resb 20  
    len resb 2  
    arr resb 2
```

```
global _start  
section .text  
_start:
```

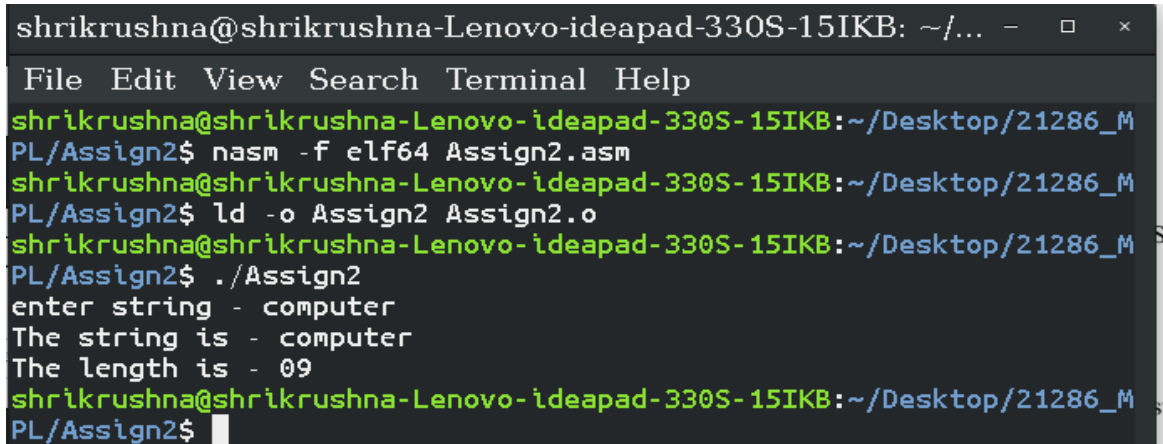
```
    mcr 1, msg1, 15  
    mcr 0, wrd, 15  
    call printByte  
    mcr 1, msg2, 16  
    mcr 1, wrd, 15  
    mcr 1, msg3, 16  
    mcr 1, arr, 2  
    mcr 1, newline, 1  
    mov rax, 60  
    mov rdi, 00  
    syscall
```

```
printByte:  
    mov rsi, arr  
    mov cl, 02  
    mov bl, 00  
l:  
    rol al, 04h  
    mov bl, al  
    and bl, 0Fh
```

```
cmp bl, 09h
jbe m
add bl, 07h
m:
add bl, 30h
mov [rsi], bl

inc rsi
dec cl
jnz l
Ret
```

OUTPUT:-

A terminal window with a dark background and light text. The title bar shows the user's name and the machine name. The terminal displays the execution of assembly code. It starts with the user running 'nasm -f elf64 Assign2.asm', followed by 'ld -o Assign2 Assign2.o', and then './Assign2'. The program outputs 'enter string - computer', 'The string is - computer', and 'The length is - 09'.

```
shrikrushna@shrikrushna-Lenovo-ideapad-330S-15IKB: ~/... - □ ×
File Edit View Search Terminal Help
shrikrushna@shrikrushna-Lenovo-ideapad-330S-15IKB:~/Desktop/21286_M
PL/Assign2$ nasm -f elf64 Assign2.asm
shrikrushna@shrikrushna-Lenovo-ideapad-330S-15IKB:~/Desktop/21286_M
PL/Assign2$ ld -o Assign2 Assign2.o
shrikrushna@shrikrushna-Lenovo-ideapad-330S-15IKB:~/Desktop/21286_M
PL/Assign2$ ./Assign2
enter string - computer
The string is - computer
The length is - 09
shrikrushna@shrikrushna-Lenovo-ideapad-330S-15IKB:~/Desktop/21286_M
PL/Assign2$
```