Department of
Computer Science and Engineering

Title: Implement Procedure in Assembly Language Programming

Microprocessors and Microcontrollers Lab

CSE 304

Green University of Bangladesh

1 Objective(s)

- To understand 8086 instructions related to Procedure using Assembly Language Program.

2 Problem analysis

With procedures we are able to write a separate piece of code, call it within our program, and return to the point that we left, having completed the code in the procedure. Procedures are also known as subroutines, functions or methods.

Call and Return Instructions

- We use the CALL instruction to transfer execution to the procedure.
- We use the RET instruction to return to where the procedure was called from.

Execution of Call instruction results-

- IP is incremented to point to the next instruction and stored (on the stack).
- The address of the first instruction in the procedure is put into IP.
- Execution is restarted in the procedure.

Execution of Return instruction results-

- The old IP is restored (from the stack).
- Execution is restarted at the point where the procedure was called from.

3 Assembly Language Program Example for Procedure

```
1  ORG 100H
2
3  .DATA
4  StrArray DB 'Hello World!!$' ; define string to display
5
6  .CODE
7  MAIN PROC
8  MOV AX, @DATA
9  MOV DS, AX
10
11 LEA DX, StrArray ; set DX to point to 1st element of string array StrArray
12 CALL USER ; call procedure
13
14 MOV AH, 4Ch
15 MOV AL, 00h ; a code after procedure call and return
16 INT 21h ; exit to DOS
17
18 MAIN ENDP
19
20 USER PROC ; declare a procedure named USER
21 MOV AH, 09h
22 INT 21h
23
24 RET ; return to MAIN procedure
25 USER ENDP ; end of procedure USER
26 END MAIN ; end of program
```

4 Sample Input/Output (Compilation, Debugging & Testing)

The program will print a string using a procedure named USER.

```
Hello World!!
```

5 Discussion & Conclusion

Based on the focused objective(s) to understand about procedure in assembly language programming, the additional lab exercise made me more confident towards the fulfilment of the objectives(s).

6 Lab Task (Please implement yourself and show the output to the instructor)

1. Write an assembly language code to take natural number series as input, and as output show the following using two different procedures:
   i. The summation of odd numbers.
   ii. The summation of even numbers.

7 Lab Exercise (Submit as a report)

- Write an Assembly Language code that takes any 5 of decimal digits (0 9) as input and calculates the average, largest and smallest of them in three different procedures and show the output like the following:

```
Input:
Enter the elements of array: 2 4 1 3 5
Output:
AVERAGE = 3
LARGEST = 5
SMALLEST = 1
```

- Write an Assembly Language code that takes any 7 of decimal digits (0 9) in any order as input and rearrange them in ascending and descending order. Use two different procedures for arranging the digits in ascending and descending order.

```
Input:
Enter the elements of array: 2 4 1 3 5 9 8
Output:
Ascending: 1 2 3 4 5 8 9
Descending: 9 8 5 4 3 2 1
```

8 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.