Rajshahi University of Engineering & Technology

Course No.: CSE 3110
Course Title: Sessional Based on CSE 3109

Submitted To:

Sadia Zaman Mishu
Assistant Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

Submitted By:

Name: Mondol Mridul Provakar
Roll: 1803062
Section: B
Department: Computer Science & Engineering

Problem No: 4

Problem Description:

Write a program that lets the user type some text, consisting of words separated by blanks, ending with a carriage return, and display the text as same word order as entered, but with the letters in each word reversed and all the vowels in capital letters.

Theory:

Registers used:

i) Accumulator (AX): For arithmetic and logical instructions.
ii) Base (BX): To hold the address of data.
iii) Counter (CX): For loops.
iv) Data (DX): To hold data for output.
v) Stack Pointer (SP): To indicate the location of the last item put onto the stack.

Instructions used:

i) MOV
ii) SUB
iii) XOR
iv) CMP
v) XCHG

vi) LOOP
vii) PUSH
viii) POP
ix) INC

The stack is a LIFO (last in, first out) data structure implemented in the RAM area and is used to store addresses and data when the microprocessor branches to a subroutine. Then the return address used to get pushed on this stack. Also to swap values of two registers and register pairs we use the stack as well.

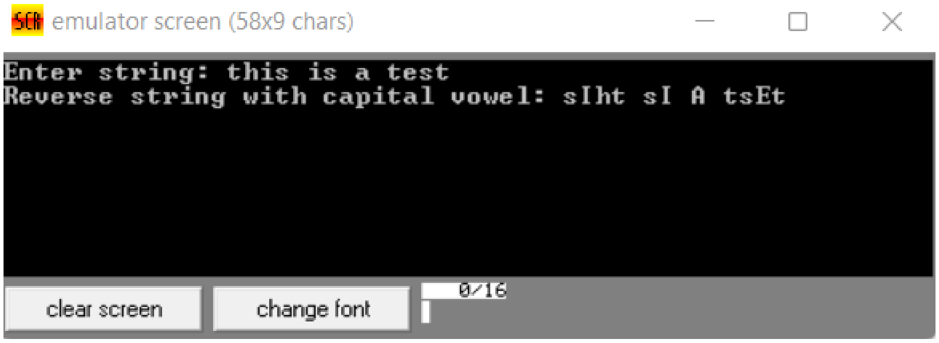
Source Code:

```
.model small

.stack 100h

.data
msg_input  db "Enter string: $"
msg_output db 10,13, "Reverse string with capital vowel: $"

.code
main proc
  mov ax, @data
  mov ds, ax
  mov ah, 9
  lea dx, msg_input
  int 21h

  xor cx, cx

reverse:
   pop dx
   xchg bx, sp
   push dx
   xchg bx, sp
   loop reverse

 show:
  mov ah, 9
  lea dx, msg_output
  int 21h

 xchg bx, sp
 mov ah, 2

 while:
   xor cx, cx

  word:
```

```
 input:
   mov ah, 1
   int 21h
   cmp al, 0dh
   je end_input
   xor ah, ah
   cmp al,'a'
   je capital
   cmp al,'e'
   je capital
   cmp al,'i'
   je capital
   cmp al,'o'
   je capital
   cmp al,'u'
   je capital
   push ax
   inc cx
   jmp input
 capital:
  sub al,32
  push ax
  inc cx
  jmp input

end_input:
 cmp cx, 0
 je show

mov bx, 50h
```

```
      cmp sp, 50h
      jge end_word
      pop dx
      cmp dx, 20h
      je end_word

      xchg bx, sp
      push dx
      inc cx
      xchg bx, sp

      jmp word

    end_word:

    xchg bx, sp

    show_word:
      pop dx
      int 21h
      loop show_word
    xchg bx, sp

    cmp sp, 50h
    jge end_while
    mov dl, 20h
    int 21h
    jmp while

  end_while:

  mov ah, 4ch
  int 21h
  main endp

end main
```

Output:



```
Enter string: this is a test
Reverse string with capital vowel: sIht sI A tsEt
```

Discussion:

A stack was used to store the string. During taking input, if a vowel was found, it was converted to uppercase letter. Then the stack was divided into two parts. At first, one by one character was popped out from the stack and pushed to the new one. Then it was popped out from the new stack till any black space found and the string became reversed.