KHULNA UNIVERSITY OF ENGINEERING AND TECHNOLOGY, KUET

SESSIONAL REPORT

Course No: CSE 2204

Department of: Computer Science and Engineering

Experiment No: 9

Name of the Experiment: To develop an assembly program that counts the number of vowel in a string

Remarks

Date of Performance: 09.05.21

Date of Submission: 19.05.21

Name: Rifat Arefin

Roll: 18070117

Year: 2nd

Semester: 2nd

No. of experiment : 09

Name of experiment : To develop an assembly program that counts the number of vowel in a string.

Objectives:

1 To find the number of vowel in a string

2 To learn about string in assembly languge

Introduction:

A string is a data type used in programming, such as an integer and floating point unit, but is used to represent next rather than numbers.

Generally we specified the length of the string by either of the two ways -

1. Explicitly storing string length

2. Using a sentinel character.

We can store the string length explicitly by using the $-location counter symbol that represent the current value of the location counter.

In the following example

```
str db "Helloworld"
len equ ($-str) ; this is the length of the
                  string.
```

Alternatively, we can also store strings with a sentinel character to delimit a string instead of storing the string length explecitly The sentinel character should be a special character that does not appear within a string.

For example,

```
massage db 'KUET CSE',0
```

Apparatus Required: emu 8086, loptop

Methodology:

Code:

```
org 100h
, data
 str db 'kuetcse'        ; given string
 n equ ($-str)           ; length of string 'str'
vow db 'aeiouAEIOU'      ; string to compare with
m  equ ($-vow)           ; length of string 'vow'
vowelcount db 0h ; initializing vowel count with
                   zero(0)

. code
mov cx, n   ; initialize cx with n
 lea si ,str ; store the ~~values~~ address of str in
               SI
loop 1 :          ; starting of loop1.
mov dx, m         ; initializing dx with m
```

```
lea di, vow          ; load address of 'vow' string to di

loop2:               ; start of loop2

      mov  al, [si]    ; copy the value of si to al

      mov  bl, [di]    ; copy the value of di to bl

      cmp  al, bl      ; comparing the value of al
                                        and bl
      je   inc_vowel   ; if ~~a~~ the value of al and bl is
      inc  di                equal then jump to inc.v

      dec  dx          ; decrease the value of dx

      cmp  dx,0h       ; comparing that dx is 0h ore
                                    not
      jne  loop2
                       ; if dx is not equal to zero
   inc  si
                             then jump to loop2.
   loop  loop1         ; increment the value of si

ret


inc_vowel:

        inc   vowelcount ; increment the value of
        inc   si                    vowel count and
        loop  Loop1              the value of si.
```

Result and discussion :

From this experiment, we had learnt about string and used it in a program. Here we tried to compare two strings. One was our given string and another one was a string ~~to~~ consists of all vowels ~~or~~ in upper case and lower case. we had compared byte by byte of a string. we used various inputs and come to a decision that our program was correct and it performed well.

Conclusion:

This experiment was very important to learn assembly language well. we learnt strin from this experiment and used string data to perform a program which helped us to

~~make~~ get cristal clear idea about string. As we took various inputs, ~~so~~ and got our expected result, so finally we can say that our program was performing well

References:

1. Microprocessor and Interfacing - D.V. Hall

2. emu 8086/ documentation/index. html.