Additional Exercise ch06

By

407261128

康智絜

X86 Assembly Language

Fall 2019

Date Submitted: November 16, 2019

**Test Plan :**

Designafinitestatemachinethat can determine if the input contains even number of 0s. The possible input sequence can be 001, 0011, 010, 1100, ...

|  |  |  |
| --- | --- | --- |
| **Test Case** | **Number** **Input** **Values** | **Expected** **output** |
| 1 | 001 | There is even number of 0s. |
| 2 | 0011 | There is even number of 0s. |
| 3 | 010 | There is even number of 0s. |
| 4 | 1100 | There is even number of 0s. |

**Feedback: (required)**

1.一開始搞不清楚FSM轉換State的方法,寫了很久才知道,原來沒有從StateA jump 到新的地方, 會一直往下執行下去, 所以StateB也會執行到

2.在寫程式之前應該要先確定自己想的FSM的diagram是正確的,不然會寫了很久卻發現FSM有bug

**Appendix A: Test Log (required)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case Number** | **Input Values** | **Actual Output** | **Result** |
| 1 | 0011 | There is even number of 0s. | True |
| 2 | 10111 | There is odd number of 0s. | True |
| 3 | 12345 | Invalid input | True |
| 4 | 12001 | Invalid input | True |
| 5 | 11111 | There are no zero here. | True |

**Appendix B: Source Code (required)**

INCLUDE Irvine32.inc

.data

Text BYTE 'Plaese input your number: ',0

TextStateOdd BYTE 'There is odd number of 0s.',0

TextStateEven BYTE 'There is even number of 0s.',0

TextNoZero BYTE 'There are no zero here.',0

WrongMsg BYTE 'Invalid input', 0

Buffer BYTE 10000 DUP(0)

byteCount DWORD ?

.code

main proc

mov edx, offset Text

call WriteString ; print 'Plaese input your number'

mov edx, OFFSET buffer ; specified buffer

mov ecx, (SIZEOF buffer)-1 ; eliminate NULL, specified max read length of string

call ReadString ; input string

mov byteCount, eax ; the string length

mov ecx, byteCount ; ecx = the string length

StateA:

cmp ecx, 0 ; if(ecx==0)

je LabelNoZero ; jump LabelNoZero

dec ecx ; the location which the word we want to cpmpare

mov al, [ Buffer + ecx ]

cmp al, '0' ; if(al=='0')

je StateC ; jump StateC

cmp al, '1' ; else if(al=='1')

je StateA ; jump StateB

mov edx, offset WrongMsg

call WriteString

call crlf

jmp ExitLabel

StateB:

cmp ecx, 0 ; if(ecx==0)

je LabelEven ; jump LabelEven

dec ecx

mov al, [ Buffer + ecx ]

cmp al, '0' ; if(al=='0')

je StateC ; jump StateC

cmp al, '1' ; else if(al=='1')

je StateB ; jump StateB

mov edx, offset WrongMsg

call WriteString

call crlf

jmp ExitLabel

StateC:

cmp ecx, 0 ; if(ecx==0)

je LabelOdd ; jump LabelEven

dec ecx

mov al, [ Buffer + ecx ]

cmp al, '0' ; if(al=='0')

je StateB ; jump StateB

cmp al, '1' ; else if(al=='1')

je StateC ; jump StateC

mov edx, offset WrongMsg

call WriteString

call crlf

jmp ExitLabel

LabelOdd:

mov edx, offset TextStateOdd ; print 'There is odd number of 0s.'

call WriteString

call Crlf

jmp ExitLabel ; end this program

LabelEven:

mov edx, offset TextStateEven ; print 'There is odd number of 0s.'

call WriteString

call Crlf

jmp ExitLabel ; end this program

LabelNoZero:

mov edx, offset TextNoZero ; print 'There are no '0' here.'

call WriteString

call Crlf

jmp ExitLabel ; end this program

ExitLabel:

exit ; exit this program

invoke ExitProcess,0

main endp

end main

**Diagram**

Odd

Even

‘0’

‘1’

‘1’

‘1’

Start

C

B

A

‘0’

‘0’