Recursive Fibonacci

Programming Assignment # 2

By
[406261597]
[林子傑]

X86 Assembly Language
Fall 2018
Date Submitted: December 22, 2018

Test Plan and TestCases (required):

用遞迴實作費式數列

Test Case Number	Input Values	Expected output
1	-1	That number was out of range, try again.
2	26	That number was out of range, try again.
3	5	F(+5)=F(+4)+F(+3)= F(+4)=F(+3)+F(+2)= F(+3)=F(+2)+F(+1)= F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+1)=+1 F(+3)=+2 F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+4)=+3 F(+3)=F(+2)+F(+1)= F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+1)=+1 F(+3)=+2 F(+5)=+5

Feedback: (optional)

原本用invoke來實作遞迴，但是因為對資料存放還不太了解，所以後來放棄了。再次了解，越強的東西越難駕馭。

Appendix A: Test Log (required)

狀況1：PrintSpace無限迴圈，會一直輸出空白

發現：為邏輯問題，如果一開始ecx會袍無限迴圈

```
PrintSpace PROC uses eax

mov al,' '

PS1:

call WriteChar

Loop PS1
```

```
ret

PrintSpace ENDP
```

解決：改用**cmp**判斷

```
PrintSpace PROC uses eax

mov al,' '

PS1:

cmp ecx,0

je PSQ

call WriteChar

dec ecx

jmp PS1

PSQ:

ret

PrintSpace ENDP
```

狀況**2**：無限迴圈

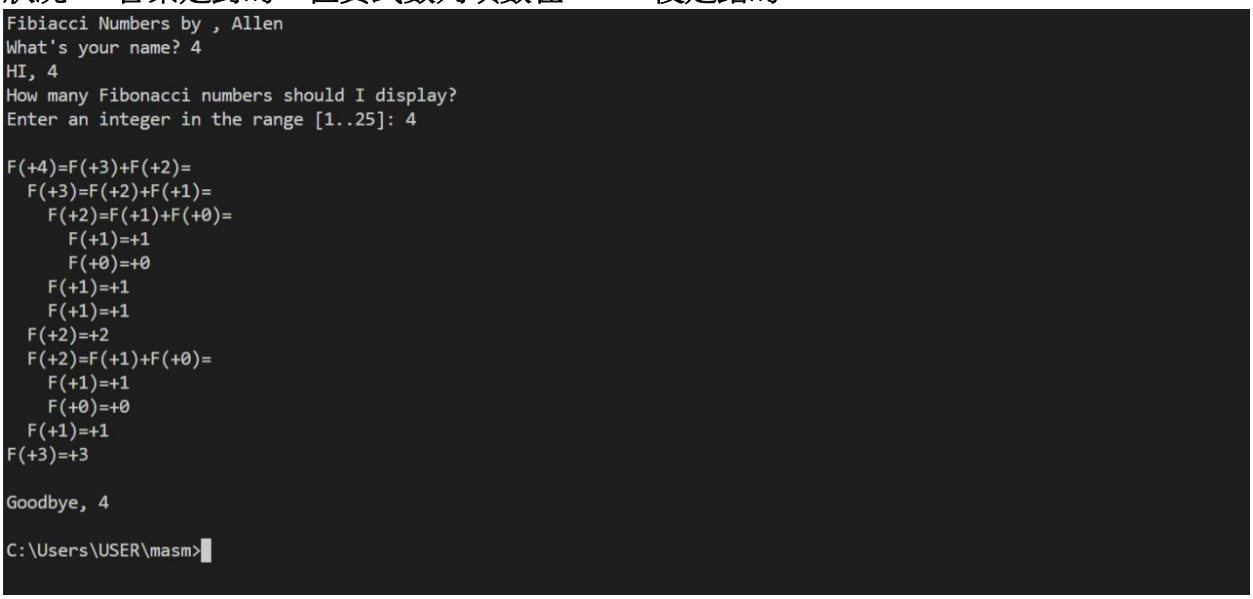
```
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
F(+249=F(+248+F(+247=
```

發現：沒設好初值

解決

```
mov num,al
```

狀況**3**：答案是對的，但費式數列項數在**return**後是錯的



解決：用**invoke**不知道資料的位置在哪，所以不用**invoke**，另外開一個變數紀錄現在為第幾項

Test Case Number	**Input Values**	**Date &Time**	**Actual Output**	**Result**
1	-1	12/22/08 19:23 pm	That number was out of range, try again.	Pass
2	26	12/22/08 19:23 pm	That number was out of range, try again.	Pass
3	5	12/22/08 19:23 pm	F(+5)=F(+4)+F(+3)= F(+4)=F(+3)+F(+2)= F(+3)=F(+2)+F(+1)= F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+1)=+1 F(+3)=+2 F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+4)=+3 F(+3)=F(+2)+F(+1)= F(+2)=F(+1)+F(+0)= F(+1)=+1 F(+0)=+0 F(+2)=+1 F(+1)=+1 F(+3)=+2 F(+5)=+5	Pass

```
Fibiacci Numbers by , Allen
What's your name? owowo
HI, owowo
How many Fibonacci numbers should I display?
Enter an integer in the range [1..25]: -1

That number was out of range, try again.
How many Fibonacci numbers should I display?
Enter an integer in the range [1..25]: 26

That number was out of range, try again.
How many Fibonacci numbers should I display?
Enter an integer in the range [1..25]: 5
How many Fibonacci numbers should I display?
Enter an integer in the range [1..25]: 5

F(+5)=F(+4)+F(+3)=
  F(+4)=F(+3)+F(+2)=
    F(+3)=F(+2)+F(+1)=
      F(+2)=F(+1)+F(+0)=
        F(+1)=+1
        F(+0)=+0
      F(+2)=+1
      F(+1)=+1
    F(+3)=+2
    F(+2)=F(+1)+F(+0)=
      F(+1)=+1
      F(+0)=+0
    F(+2)=+1
  F(+4)=+3
  F(+3)=F(+2)+F(+1)=
    F(+2)=F(+1)+F(+0)=
      F(+1)=+1
      F(+0)=+0
    F(+2)=+1
    F(+1)=+1
  F(+3)=+2
F(+5)=+5

Goodbye, owowo
```

Appendix B: Source Code (required)

```
; .386


Include \masm32\include\Irvine32.inc

Includelib \masm32\lib\Irvine32.lib

includelib \masm32\lib\Kernel32.lib

includelib \masm32\lib\User32.lib


.data

num BYTE 0

str_1 BYTE "Fibiacci Numbers by , Allen",0

str_2 BYTE "What's your name? ",0

str_3 BYTE "HI, ",0

str_4 BYTE "How many Fibonacci numbers should I display?",0

str_5 BYTE "Enter an integer in the range [1..25]: ",0

str_6 BYTE "That number was out of range, try again.",0

str_7 BYTE " ",0

str_8 BYTE "Goodbye, ",0

N BYTE 10 DUP(?)

non BYTE 0

tmp DWORD 0

F MACRO BYTE

mov al,'F'

call WriteChar

mov al,'('

call WriteChar

movzx eax,BYTE
```

```
call writeInt
mov al,')'
call WriteChar
ENDM

.code

main PROC
L1: ; print info
call Clrscr
mov edx,OFFSET str_1
call WriteString
call Crlf
mov edx,OFFSET str_2
call WriteString
mov edx,OFFSET N
mov ecx,10
call readString
mov edx,OFFSET str_3
call WriteString
mov edx,OFFSET N
call WriteString
call Crlf
L2: ; chack range
mov edx,OFFSET str_4
call WriteString
```

```
call Crlf
mov edx,OFFSET str_5
call WriteString
call readint
call Crlf
cmp eax,1
jl L3
cmp eax,25
jg L3
jmp L4
L3: ; print error messenge
mov edx,OFFSET str_6
call WriteString
call crlf
jmp L2
L4: ; print fibonacci
mov non,al
push eax
xor ebx,ebx
call fib
add esp,4
L5: ; print goodbye
call crlf
mov edx,OFFSET str_8
call WriteString
mov edx,OFFSET N
```

```
call WriteString
call Crlf
exit
main ENDP

Fib PROC
push ebp
mov ebp,esp
sub esp,4 ; space for local Dword [ebp-8]
mov eax,[ebp+8]
cmp eax,2
jl Base

mov ecx,ebx
call PrintSpace
mov ecx,eax
call PrintInfo1
add ebx,2

; fib(n-1)
dec non
dec eax
push eax
call fib

mov [ebp-4],eax ; store firsr result.
```

```
.
; fib(i-2)
dec non
dec DWORD PTR [esp]
call fib
add non,2

add DWORD PTR [esp],2
sub ebx,2
add esp,4 ; clear stack
add eax,[ebp-4] ; eax=fib(i-1)+fib(i-2)
mov ecx,ebx
call PrintSpace
mov tmp,eax
call PrintInfo2
jmp Quit
Base:
mov ecx,ebx
call PrintSpace
mov tmp,eax
call PrintInfo2
Quit:
mov esp,ebp
pop ebp
ret
Fib ENDP
```

```
PrintSpace PROC uses eax ; print space
mov al,' '
PS1:
cmp ecx,0
je PSQ
call WriteChar
dec ecx
jmp PS1
PSQ:
ret
PrintSpace ENDP

PrintInfo1 PROC uses eax ;print f(x)=f(x-1)+f(x-2)=
F cl
mov al,'='
call WriteChar
dec cl
F cl
mov al,'+'
call WriteChar
dec cl
F cl
mov al,'='
call WriteChar
call Crlf
```

```
ret

PrintInfo1 ENDP


PrintInfo2 PROC uses eax ;print f(x)=

F non

mov al,'='

call WriteChar

mov eax,tmp

call writeInt

call Crlf

ret

PrintInfo2 ENDP


END main
```