

## Experiment 4

**Class:** SE Comp

**Year:** 2020-21

**Performed by:** Danyl Fernandes, 72

### Fibonacci series:

#### **Code:**

```
org 100h
.data
    count dw 0ah
    res db 10 dup(?)
.code
    mov ax, @data
    mov ds, ax
    mov cx, count
    mov si, offset res
    mov al, 00h
    mov bl, 01h
    mov [si], al
    inc si
    mov [si], bl
    inc si
up:
    mov al, [si-2]
    mov bl, [si-1]
    add al, bl
    mov [si], al
    inc si
    loop up
ret
```

## Output:

emu8086 - assembler and microprocessor emulator 4.08

file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help about

```
01 org 100h
02 .data
03 count dw 0ah
04 res db 10 dup(?)
05 .code
06 mov ax, @data
07 mov ds, ax
08 mov cx, count
09 mov si, offset res
10 mov al, 00h
11 mov bl, 01h
12 mov [si], al
13 inc si
14 mov [si], bl
15 inc si
16 up:
17 mov al, [si-2]
18 mov bl, [si-1]
19 add al, bl
20 mov [si], al
21 inc si
22 loop up
23 ret
24
25
26
27
28
29
```

emulator: noname.com\_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	07	59
BX	00	37
CX	00	00
DX	00	00
CS	F400	
IP	0154	
SS	0700	
SP	FFFA	
BP	0000	
SI	0110	
DI	0000	
DS	0700	
ES	0700	

F400:0154

Address	Hex	Dec	Comment
F4154	CF 207	±	I RET
F4155	00 000	NULL	ADD [BX + SI], AL
F4156	00 000	NULL	ADD [BX + SI], AL
F4157	00 000	NULL	ADD [BX + SI], AL
F4158	00 000	NULL	ADD [BX + SI], AL
F4159	00 000	NULL	ADD [BX + SI], AL
F415A	00 000	NULL	ADD BH, BH
F415B	00 000	NULL	DEC BP
F415C	00 000	NULL	SBB CL, BH
F415D	00 000	NULL	ADD [BX + SI], AL
F415E	00 000	NULL	ADD [BX + SI], AL
F415F	00 000	NULL	ADD [BX + SI], AL
F4160	FF 255	RES	ADD [BX + SI], AL
F4161	FF 255	RES	ADD [BX + SI], AL
F4162	CD 205	=	ADD BH, BH
F4163	1A 026	→	DEC BP
F4164	CF 207	±	ADD BH, CL
F4165	00 000	NULL	ADD [BX + SI], AL
F4166	00 000	NULL	ADD [BX + SI], AL
F4167	00 000	NULL	...
F4168	00 000	NULL	...
F4169	00 000	NULL	...

screen source reset aux vars debug stack flags

Random Access Memory

F400:0154 update table list

Address	Hex	Dec	Comment
F400:0154	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 1A	±	...
F400:0164	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 00	±	...
F400:0174	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 04	±	...
F400:0184	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 10	±	...
F400:0194	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 12	±	...
F400:01A4	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 13	±	...
F400:01B4	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 16	±	...
F400:01C4	CF 00 00 00 00 00 00 00-00 00 00 00 FF FF CD 14	±	...

## Conclusion:

We successfully wrote an assembly language program to find the fibonacci series of N given terms

## Exp 04

Aim: To write an assembly language program to find fibonacci series of  $N$  given terms.

### Algorithm:

- Start the program
- Initialize the data section
- Initialize the counter = 10, 0AH
- Initialize SI to the starting address of `dec`
- Store the first term at location where SI is pointing  $\Delta$  increment SI to point next location
- Store next term '1' at location where SI is pointing  $\Delta$  increment SI to point next location
- Decrement counter & check if count = 0, if not go to step 8
- Stop

## Flow Chart:

