8 bit

+53 → 110101 → 00110101

+13 → 1101 → 4_bit → 0011

2's Comp

1) Divisor ≠ 0 ✓

2) Divisor > Dividend ✓ ↓ 8 بیشتر

درغیر overflow

$$\begin{array}{r|l} 53 & 13 \\ \hline 52 & 4 \\ \hline 1 & \end{array}$$

4 → خارج قسمت

1 ← باقی مانده

		A	Q
		0 0 0 1 1	1 1 0 1
①	shf	0 0 1 1 0	1 0 1 0
	sub	1 0 0 1 1	
	neg	1 1 0 0 1	
	add	0 1 1 0 1	
		0 0 1 1 0	
②	shf	0 1 1 0 1	0 1 0 0
	sub	1 0 0 1 1	
	pos	0 0 0 0 0	0 1 0 1
③	shf	0 0 0 0 0	1 0 1 0
	sub	1 0 0 1 1	
	neg	1 0 0 1 1	
	add	0 1 1 0 1	
		0 0 0 0 0	
④	shf	0 0 0 0 1	0 1 0 0
	sub	1 0 0 1 1	
	neg	1 0 1 0 0	
	add	1 1 0 1	
		0 0 0 0 1	

نادرست +A

باقی مانده +1

2) -43.625 $\quad N = (-1)^S \; 1.nnn\ldots n \times 2^E$

$S = 1 \quad 43.625 \Rightarrow 101011.101$

$= 1.01011101 \times 2^5$

$E = 127 + 5 = 132 \Rightarrow 10000100$

S E M

32

1 10000100 01011101 00…0

S E M 15

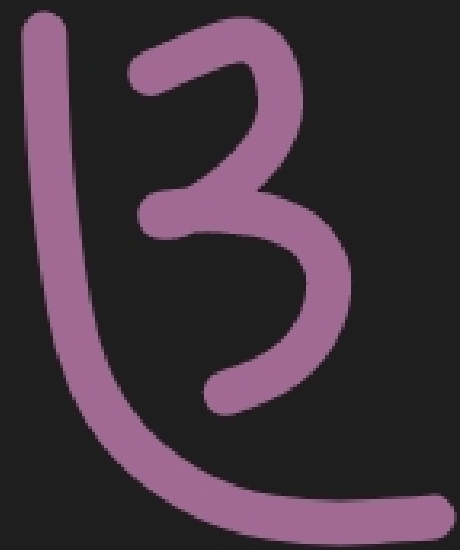
$1 + 8 + 8 + 15 = 17 + 15 = 32$ ✓

```c
#include<stdio.h>


int Fib(int n){
    int a=1;
    int b=1;
    for(int i = 2 ; i < n ; i++){
        int c = a + b;
        a = b;
        b = c;
    }

    return b;
}
int main(){


/*printf("%d" , Fib(4));*/
return 0;
}
```

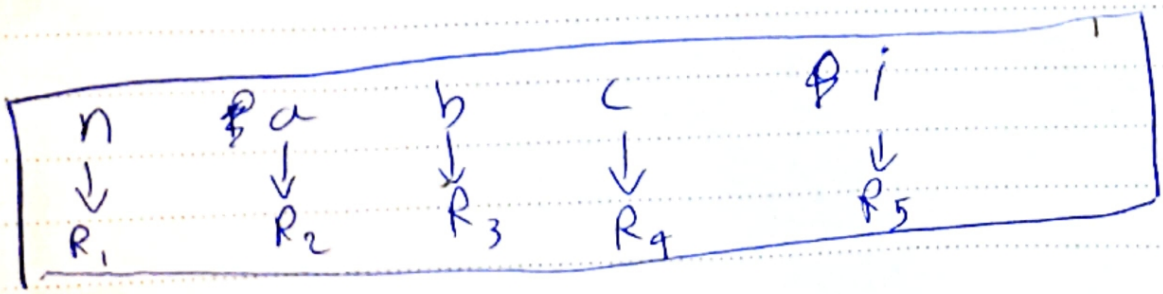


کد اسمبلی MIPS:

```
lw   R1, -4(R0)        ------> load n
addi R2, 1, R0         a = 1
addi R3, 1, R0         b = 1
addi R5, 2, R0         i = 2   initial value

loop: slt  R6, R5, R1      i < n
      beq  R6, R0, return  i ≥ n -> return
      add  R4, R3, R2      c = a+b
      add  R2, R3, R0      a = b
      add  R3, R4, R0      b = c
      addi R5, R5, 1       i++

return: jr R31             return
```



(4)

sw++ مسیر آبی برای دستور

wai مسیر قرمز برای دستور

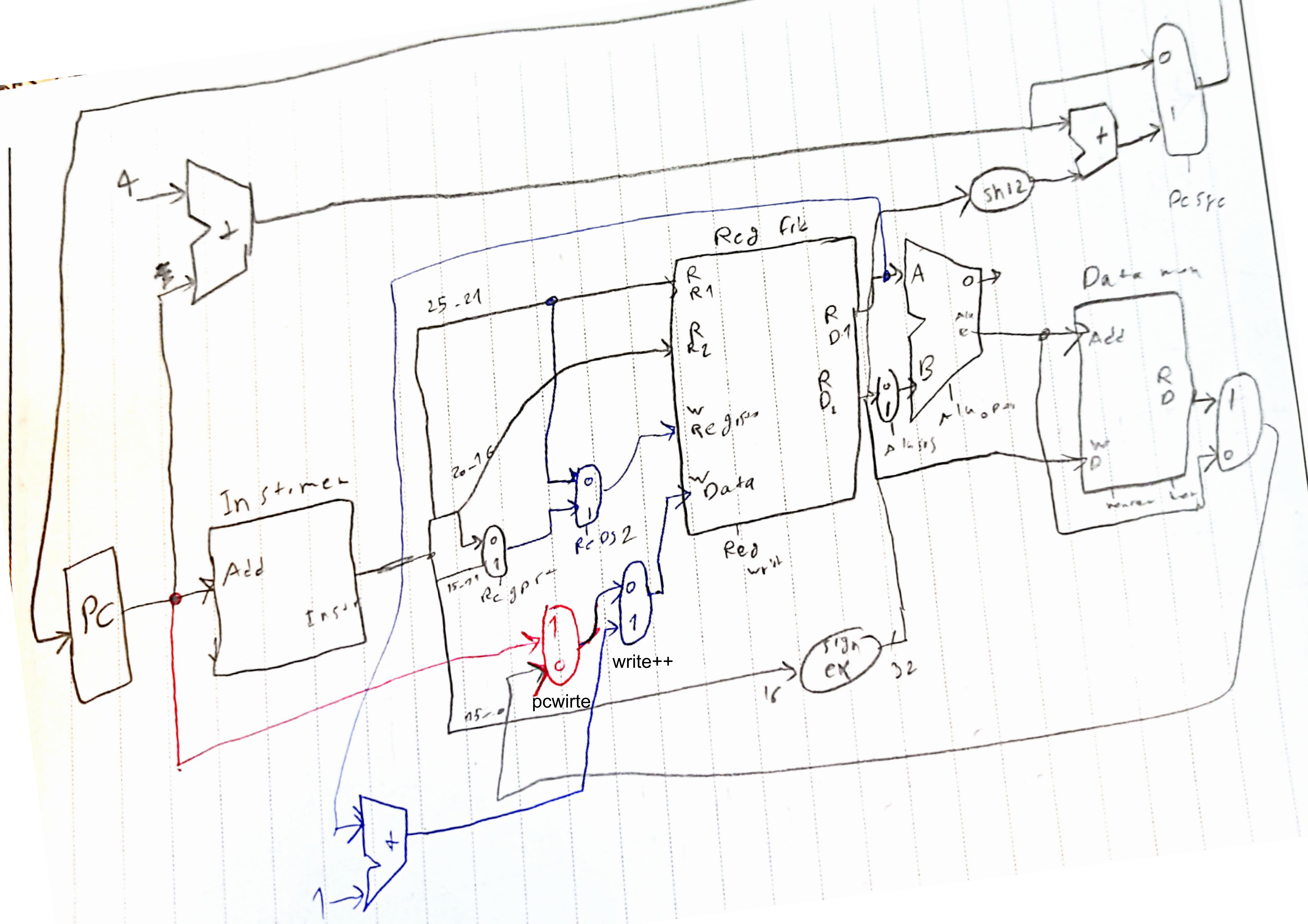
	Reg Dst	Reg Dst 2	PC write	write++	Reg write	Alu src	ALU op	PC src	mem Read	mem write	Reg Dst
sw++	—	0	—	1	1	1	00(+)	0	0	1	—
wai	—	0	1	0	1	—	—	0	0	0	—

sw++ قالب دستور

op	rs Rs	rt	adr
31–26	25–21	20–16	15

wai قالب دستور

opc	R1	port core
31–26	25–21	20–0



4
PC
Add
Inst
25-21
20-16
15-11
15-0
A
B
Add
R D
sh12
16
32
write++
pcwirte
1