810199xxx
810199xxx

1- تابع فیبوناچی در زبان C:

```c
int fib(int n)
{
    int a = 1, b = 1;
    for (int i = 2; i < n; ++i)
    {
        int c = a + b;
        a = b;
        b = c;
    }
    return b;
}
```

تابع فیبوناچی در زبان اسمبلی MIPS:

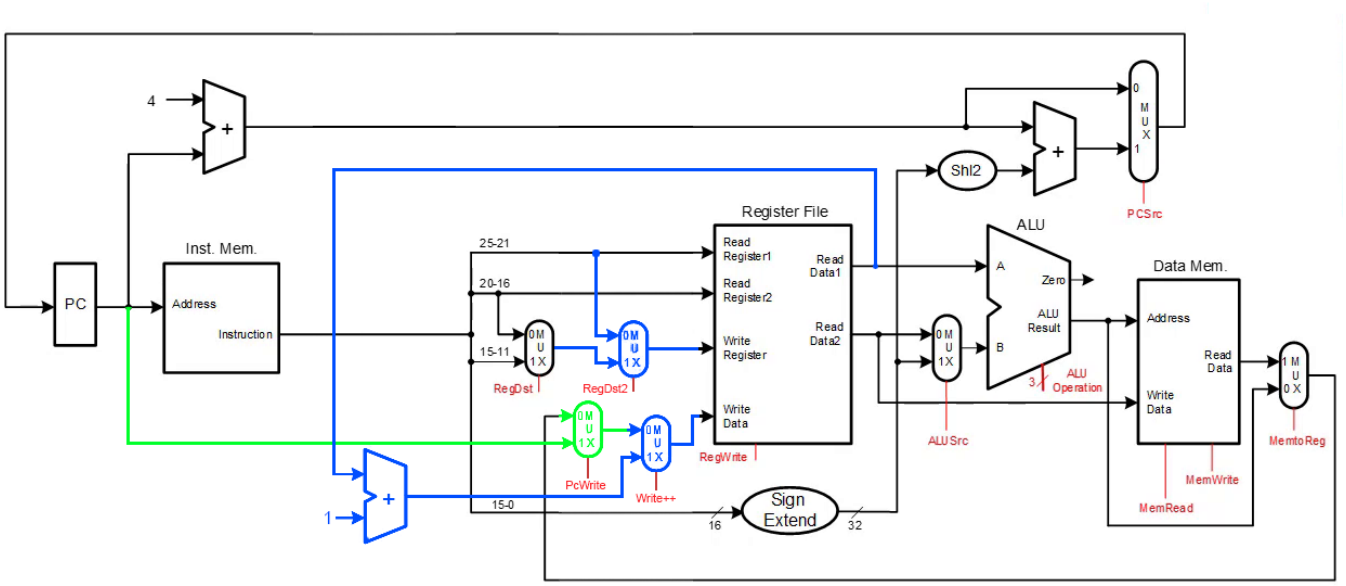
```
fib:
# Assuming that n is pushed to the stack
# and that the function returns the result in R3
lw      R1, -4(R29)         # Load n from stack into R1

addi    R2, R0, 1           # a = 1
addi    R3, R0, 1           # b = 1
addi    R4, R0, 2           # i = 2 (loop variable)

loop:
slt     R6, R4, R1          # if i < n, then goto loop
beq     R6, R0, end_fib     # if i ≥ n, then goto end_fib
add     R5, R2, R3          # c = a + b
add     R2, R3, R0          # a = b
add     R3, R5, R0          # b = c
addi    R4, R4, 1           # i++
j       loop

end_fib:
jr      R31                  # return

# ————————————————————————————————————————————————————————————

main:
# Assuming that n is stored in R10 and should be pushed to the stack
addi    R29, R29, 4         # Update stack pointer
sw      R10, 0(R29)         # Push n to stack

addi    R29, R29, 4         # Update stack pointer
sw      R31, 0(R29)         # Save return address

jal     fib                 # Jump to fib and save position to R31
lw      R31, 0(R29)         # Load return address from stack
addi    R29, R29, -8        # Update stack pointer to pop n and return address from stack
```

2- موارد الف و ب در مسیر داده شکل زیر ادغام شده‌اند. مورد الف با رنگ آبی و مورد ب با رنگ سبز نمایش داده شده است.

قالب دستور wai به صورت زیر انتخاب شده است:

OPC	R_t	Don't Care

31 26 25 21 20 0



کنترلر مربوط به دستور sw++:

RegDst	RegDst2	PcWrite	Write++	RegWrite	ALUSrc	ALUOp	PCSrc	MemRead	MemWrite	MemToReg
—	0	—	1	1	1	00 (+)	0	0	1	—

کنترلر مربوط به دستور wai:

RegDst	RegDst2	PcWrite	Write++	RegWrite	ALUSrc	ALUOp	PCSrc	MemRead	MemWrite	MemToReg
—	0	1	0	1	—	—	0	0	0	—