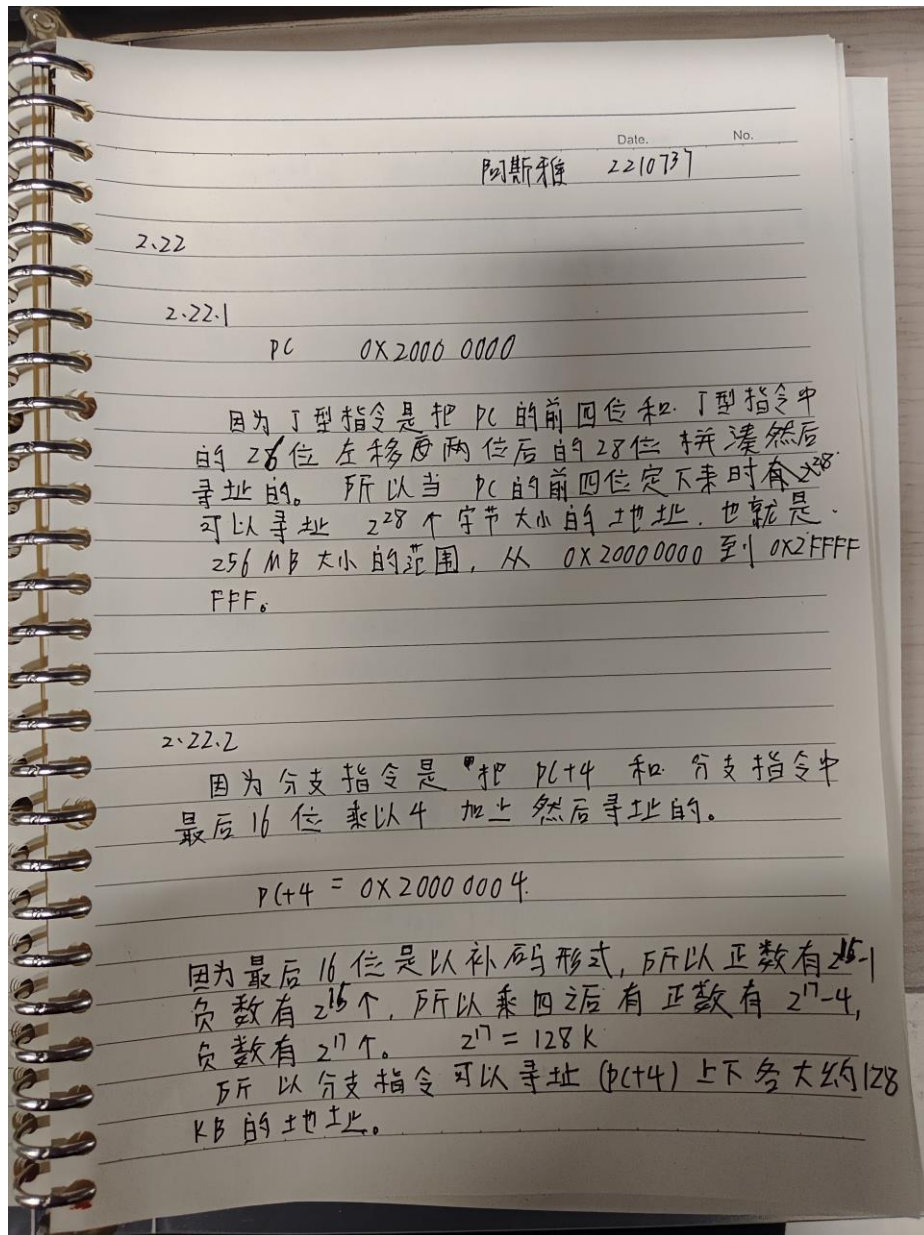


2.22



2.29

2.29

```
int fib(int n) {
    if (n == 0)
        return 0;
    else if (n == 1)
        return 1;
    else
        return fib(n-1) + fib(n-2);
}
```

因为参数 n 保存在 $\$a0$ 里面， fib 函数里面也有递归所以要用栈来解决：

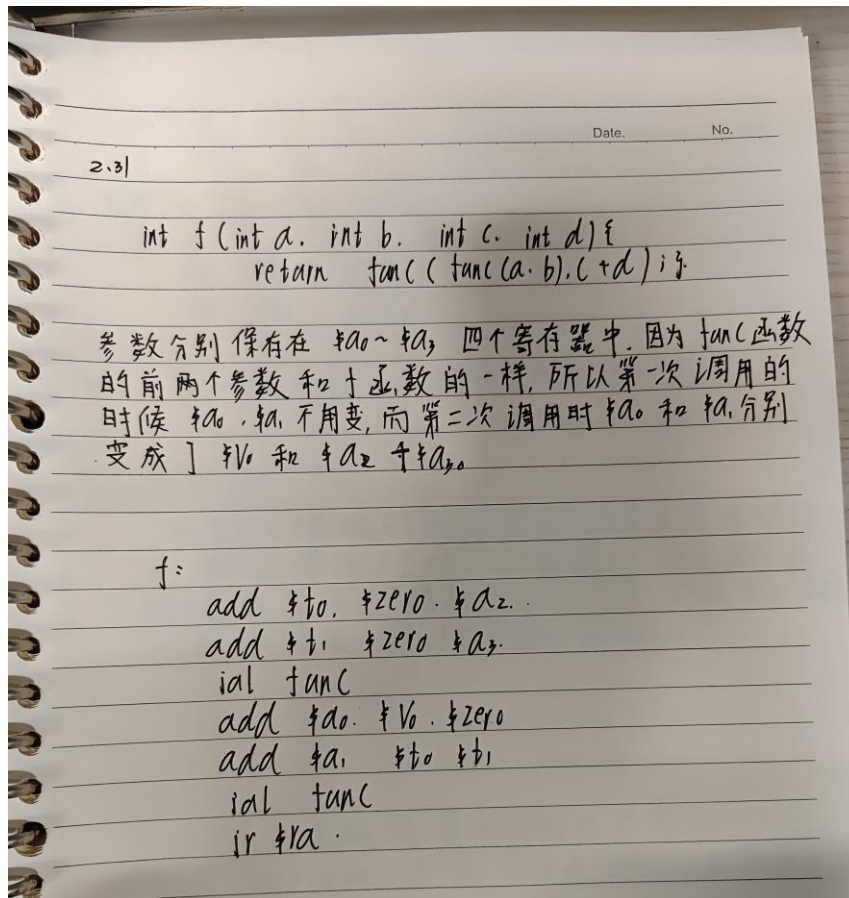
```
add $s0, $zero, $zero
fib: addi $sp, $sp, -12
sw $a0, 0($sp)
sw $s0, 4($sp)
sw $ra, 8($sp)
bne $a0, $zero, return1
add $v0, $zero, $zero
jal retN

return1: add $t0, $t0, 1
bne $a0, $t0, return2
addi $v0, $zero, 1
jal retN

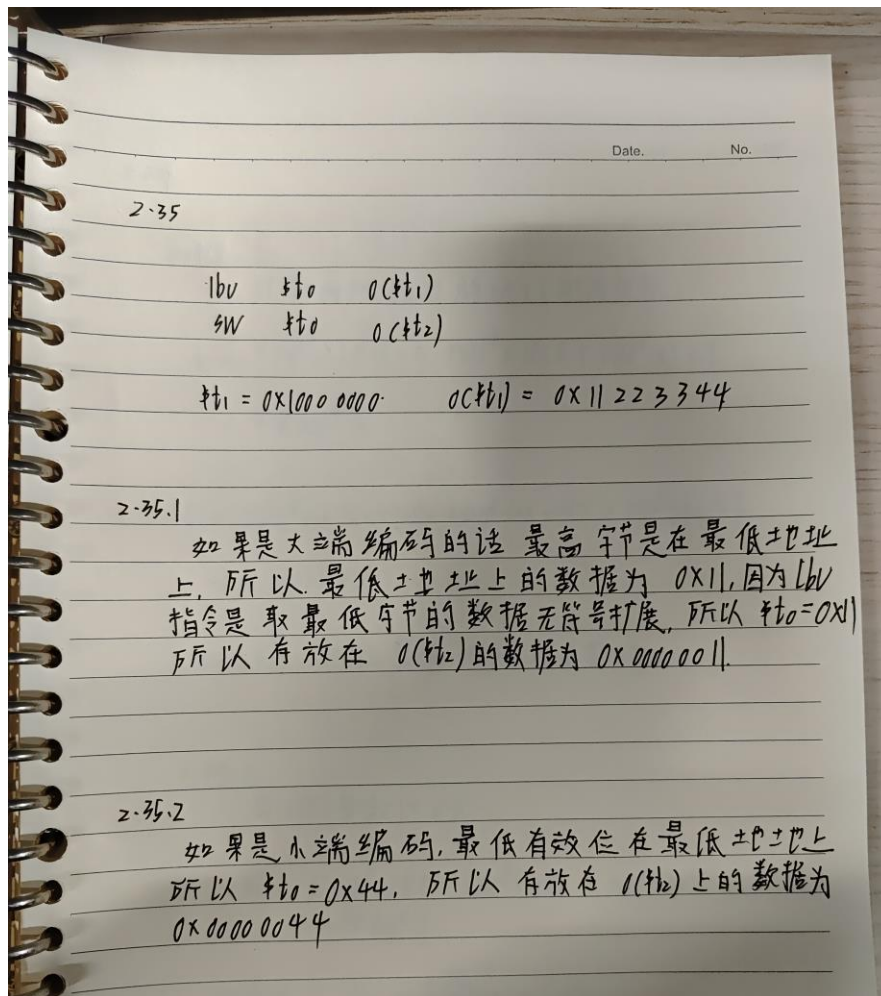
return2: addi $a0, $a0, -1
jal fib
add $s0, $v0, $zero
addi $a0, $a0, -1
jal fib
```

```
add $v0, $v0, $s0
retN: lw $a0, 0($sp)
lw $s0, 4($sp)
lw $ra, 8($sp)
addi $sp, $sp, 12
jr $ra
```

2.31



2.35



2.39

2.39

Date.

No.

2.39.1.

计算两种情况的执行时间即可.

$$t_1 = 5 \times 10^8 \times 1 \times T + 3 \times 10^8 \times 10 \times T + 1 \times 10^8 \times 3 \times T \\ = 3.8 \times 10^9 T.$$

$$t_2 = 5 \times 10^8 \times 0.75 \times 1 \times T + 3 \times 10^8 \times 10 \times 1.1 T + 1 \times 10^8 \times 3 \times 1.1 T \\ = 4.0425 \times 10^9 T$$

∵ $t_1 < t_2$ 所以整体性能没有变好.

2.39.2

① CPI(整数) = 0.5

$$t_0 = 5 \times 10^8 \times 0.5 T + 3 \times 10^8 \times 10 \times T + 1 \times 10^8 \times 3 \times T \\ = 3.55 \times 10^9 T$$

$$\text{加速比} = \frac{t_1}{t_0} = \frac{3.8 \times 10^9}{3.55 \times 10^9} = 1.07$$

② CPI(整数) = 0.1

$$t_1 = 5 \times 10^8 \times 0.1 T + 3 \times 10^8 \times 10 T + 1 \times 10^8 \times 3 T \\ = 3.35 \times 10^9 T$$

$$\text{加速比} = \frac{t_1}{t_0} = \frac{3.8}{3.35} \approx 1.13$$

如果没有比例下标寻址的方法，我们要想访问数组的某一个位置上的元素，就必须要把字地址转换为字节地址，也就是要利用左移指令，而当有了比例下标寻址，我们就可以不用左移指令，直接表示成 $B[t]$ ，因为指令可以自己帮我们做 $\times 4$ 的操作。所以 2.4 的代码可以改为：

```
add $t0, $zero, $s6[$s0].  
addi $s0, $s0, 1.  
add $t1, $zero, $s6[$s0]  
add $t0, $t0, $t1  
sw $t0, $s7[$s0]
```