一、熟悉MARS

1..data表明接下来的数据段将存储于内存的data部分中的可存储部分；.word后续的一行的内容将会以32位形式存储在字边界上；.text表明接下来的数据段将存储于内存的text部分中的可存储部分；

2.在Execute栏，最左侧有Bkpt栏，对应Source勾选即可在对应行数设置断点。指令的地址为0x00400020，第14行会执行；

3.可以执行Run->Go（F5）继续执行，Run->Step（F7）可以单步执行；

4.程序输出的数字是34，是第9个斐波那契数；

5.存储在Data段0x10010010处；

6.在Data Segment的0x10010000的Value（+10）处将0x00000009改为0x0000000d；

7.在第11行设置断点，修改$t3寄存器的值，将0x00000009改为0x0000000d，再解除断点即可；

8.它调用了一个system call，显示$v0寄存器中的值。第19行输出第n个斐波那契数的数字，第21行return，main函数结束。用法：先修改$v0寄存器中的值，在下一行调用syscall。

二、将C编译为MIPS

1.

$L3:

sw $4,0($3) //将source复制到dest

lw $4,0($2)

addiu $3,$3,4

addiu $2,$2,4

bne $4,$0,$L3

nop

此段为main函数中的for循环，sw $4,0($3) //将source复制到dest，后续为循环判断。

2.

存储位置：dest指针在.end main后中.comm dest,40,4中，.globl source指明了source指针，后续指明了对应的数组、大小与数字；

解释：

lui $3,%hi(dest)

lui $2,%hi(source+4)

addiu $3,$3,%lo(dest)

addiu $3,$3,%lo(source+4)

初始化并存入$2,$3后，在$L3循环中通过

addiu $3,$3,4

addiu $3,$3,4

进行字段移动取数据

三、函数调用的过程

#prologue

### YOUR CODE HERE ###

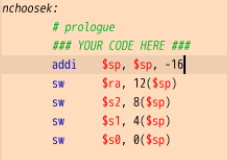
addi $sp, $sp, -16

sw $ra, 12($sp)

sw $s2, 8($sp)

sw $s1, 4($sp)

sw $s0, 0($sp)



#epilogue

### YOUR CODE HERE ###

lw $s0, 0($sp)

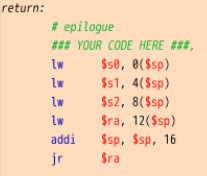
lw $s1, 4($sp)

lw $s2, 8($sp)

lw $ra, 12($sp)

addi $sp, $sp, 16

jr $ra



输出：

