

1. LOOP: `slt $t2, $0, $t0`  
`bne $t2, $0, ELSE`  
`j DONE`

ELSE: `addi $s2, $s2, 2`  
`addiu $t0, $t0, 1`  
`j LOOP`

DONE:

000000000000010000010100000001010  
 00010101000000000000000000000001  
 0000100000010000000000000000010  
 010001001010010000000000000000010  
 00100101000100000000000000000001  
 0000100000010000000000000000000

2. `la $s0, 268435456`

`andi $s0, $s0, 16711680`

3. a fib: `addi $sp, $sp, -8`  
`sw $ra, 4($sp)`  
`sw $a0, 0($sp)`  
`add $s0, $0, $a0`  
`bne $s0, $0, elseif`  
`add $v0, $0, $0`  
`addi $sp, $sp, 8`  
`jr $ra`

elseif: `addi $t0, $0, 1`  
`bne $s0, $t0, else`  
`addi $v0, $0, 1`  
`addi $sp, $sp, 8`  
`jr $ra`

else: `addi $a0, $a0, -1`  
`jal fib`  
`addi $a0, $a0, -1`  
`jal fib`  
`lw $a0, 0($sp)`  
`la $ra, 4($sp)`

`addi $sp, $sp, 8`  
`jr $ra`

For  $n=5$ , there need 200 instructions

`addi $t0, $0, 0`  
`< add $v0, $t0, $t0`

`< add $v0, $t0, $t0`

Pan Chongdon

516370910121

4. For a fib function, it can't be inline because it's recursive

5.

$0x7fffffff c \rightarrow 0x7fffffff 4 \rightarrow \$ra(4) \rightarrow \$ra(3)$   
 $0x7fffffff 8 \quad \$ra(5)$   
 $0x7fffffff c \quad 5$   
 $\begin{matrix} 4 \\ \$ra(5) \\ 5 \\ \$ra(5) \\ 5 \end{matrix} \quad \begin{matrix} 3 \\ \$ra(4) \\ 4 \\ \$ra(5) \\ 5 \end{matrix}$

$\begin{matrix} \$ra(2) \\ 0 \\ \$ra(1) \\ 1 \\ \$ra(2) \\ 2 \\ \$ra(3) \\ 3 \\ \$ra(4) \\ 4 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(1) \\ 1 \\ \$ra(3) \\ 3 \\ \$ra(4) \\ 4 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(2) \\ 2 \\ \$ra(4) \\ 4 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(2) \\ 2 \\ \$ra(4) \\ 4 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(3) \\ 3 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(3) \\ 3 \\ \$ra(5) \\ 5 \end{matrix}$

$\begin{matrix} \$ra(0) \\ 0 \\ \$ra(1) \\ 1 \\ \$ra(2) \\ 2 \\ \$ra(3) \\ 3 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \begin{matrix} \$ra(1) \\ 1 \\ \$ra(3) \\ 3 \\ \$ra(5) \\ 5 \end{matrix} \rightarrow \$ra(5) \rightarrow 5$   
5 is always at  $7fffffff c$



b. `addi $sp, $sp, -4`  
`sw $a0, 0($sp)`  
`add $v0, $0, $0`

Loop: `lw $s0, 0($a0)`  
`beq $s0, $0, END`  
`addi $t0, $0, 48`  
`slt $t1, $t0, $s0`  
`beq $t1, $0, ND`  
`addi $t0, $0, 58`  
`slt $t1, $s0, $t0`  
`beq $t1, $0, ND`  
`addi $t0, $t0, 10`  
`mult $w, $t0`  
`add $v0, $w, $s0`  
`addi $a0, $a0, 4`  
`j Loop`

ND: `addi $w, $0, -1`

END:

7. `0x00000011`

8. `2`

9. `1, 1000`

10. The immediate field will lose 4 bits and for rs and rt there will be 2 more bits

1. `beq $s1, $s2, 10, if[$s1=$s2] PC=PC+4+4x10, I format`