addiu $t1, $zero,10

re: addi $t1, $t1,-1

bne $t1,$zero,re

24009000a

2129ffff

1120fffe

main:

addi $a0, $zero, 5 # let the argument be 5 00

jal fact # jump to fib label, like calling F(5) 04

j exit 08

fact:

addi $sp, $sp, -8 # allocate 8 bytes to this stack 0c

sw $ra, 0($sp) # save return address in stack 10

sw $a0, 4($sp) # save argument value in stack 14

slti $t0, $a0, 1 # if n < 1 then $t0 = 1, else $t0 = 0 18

beq $t0, $zero, L1 # if $t0 == 0 then jump to branch L1 1c

addi $v0, $zero, 1 # let $v0 be 1 20

addi $sp, $sp, 8 # let $sp point to upper stack 24

jr $ra # jump to the next line of the line calling fib 28

L1:

addi $a0, $a0, -1 # n = n - 1 2c

jal fact # jump fact again, like as returning F(n - 1) 30

lw $a0, 4($sp) # recover the value of argument 34

mul $v0, $a0, $v0 # $v0 \*= $a0, like as F(n) = n \* F(n - 1) 38

lw $ra, 0($sp) # recover return address 40

addi, $sp, $sp, 8 # let $sp point to upper stack 44

jr $ra # jump to the next line of the line calling L1 48

exit: 4c

0x20040005

0x0C000003

0x08000013

0x23BDFFF8

0xAFBF0000

0xAFA40004

0x28880001

0x11000003

0x20020001

0x23BD0008

0x03E00008

0x2084FFFF

0x0C000003

0x8FA40004

0x00440018

0x00001012

0x8FBF0000

0x23BD0008

0x03E00008

DATA IN MEMORY

; str

0a48656c

6c6f2057

6f726c64

210a0000

\nHello World!\n

addi $sp,$sp,0xfffffff2

addi $t2,$zero,0x00000000

addi $t1,$zero,0x0000000e

re: lb $s1, 0x0($sp)

addi $sp,$sp,0x00000001

addi $s1,$s1,0x00000001

addi $t2,$t2,0x00000001

sb $s1,0x0($t2)

bne $t2,$t1,0xfffffffa //j 0x0000 000c

23BDFFF2

200A0000

2009000E

83B10000

23BD0001

22310001

214A0001

A1510000

1549FFFA