

10/30/2023

Noah Semashkevich

COSC 211 Lab 4 Written

① a)

.data

my-global: .word 100

.text

.global main

main:

la \$a0, my-global

lw \$t0, my-global

lw \$t1, 0(\$t0)

addi \$t1, \$t1, 1 #my-global += 1;

sw \$t1, 0(\$t0)

add \$a0, \$zero, \$t3 #t3 = x

jal leaf_function

add \$t4, \$zero, \$v0

li \$v0, 10 #exit

syscall

leaf_function:

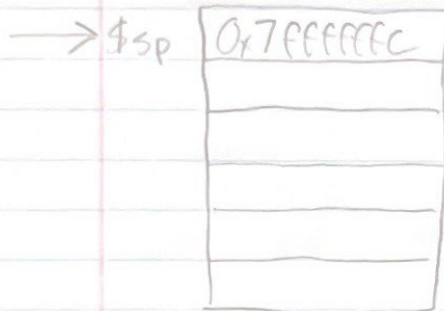
add \$s0, \$zero, \$a0

addi \$s0, \$zero, 1

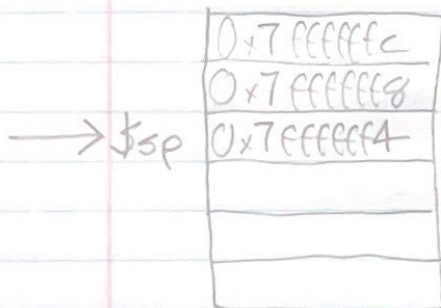
add \$v0, \$zero, \$s0

jr \$ra

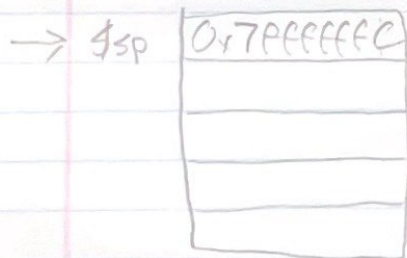
b) The stack pointer starts at $0x7ffffffc$.
Before the procedure is called the stack looks like



After the procedure call, space will be created to store \$ra and \$a0, so the stack needs to store two items



$0x7ffffff8$ will store \$a0, and $0x7ffffff4$ will store \$ra. When control returns back to the caller function, stack will load these values and stack will look like



Stack looks how it did at start of function

c)

leaf_function

addi \$sp, \$zero, -4
sw \$t0, 0(\$sp)

add \$t0, \$zero, \$a0
addi \$t0, \$t0, 1
add \$v0, \$zero, \$t0

lw \$t0, 0(\$sp)
addi \$sp, \$sp, 4
jr \$ra

② a) hello world

0x68 0x65 0x6C 0x6C 0x6F 0x77 0x6F 0x72 0x6C 0x64

= 0x68 0x65 0x6C 0x6C 0x6F 0x77 0x6F 0x72 0x6C 0x64 0x00
null terminator

b) 0123456789

= 0x30 0x31 0x32 0x33 0x34 0x35 0x36 0x37 0x38 0x39 0x00