MIPS Functions

- There is no one calling convention for functions
- Reading online you will find many conventions for calling functions

2 actors: caller and callee

- The caller calls the callee

2 types of functions

- Leaf function
    - Does not call other functions
    - May or may not use the stack
- Non-leaf function
    - Calls at least one other function
    - Must use the stack

Jumping

- jal instruction
  - jump and link
  - Stores the address to jump back to in $ra (return address)
- To jump back to the return address, callee calls j $ra at the end of the function

Non-leaf Functions

- Since the function will call another function, certain items must be stored on the stack
  - Registers to be saved ($s0 through $s7)
  - Incoming arguments ($a0 through $a3)
  - $ra
  - Local data
  - $fp - frame pointer, stores the address of the beginning of the stack frame
  - Padding - stack must be aligned to 8 byte addresses (due to doubles)

The stack

- Grows downwards (the "top" of the stack has the lowest address of any word in the stack)
- Pushing items onto the stack requires decrementing $sp
- Popping items off of the stack requires incrementing $sp
- Access items on the stack using an offset from $sp

```
lw $t0, 8($sp)  - loads the word at address $sp + 8
```