

# Subroutines

- Code [jumps](#) to the subroutine, unlike macros which use in place replacement
- Subroutines are activated in a last called first finished

## Simple Calling Conventions

- Certain registers should be used for certain things

### Subroutine Call (done by the caller)

1. Push onto stack any registers `$t0-$t9` that contain values that must be saved
2. Put argument values into `$a0-a3`
3. Call the subroutine using `jal`

## Nested Subroutines

- We know that to return to the caller, a subroutine must have the correct return address in `$ra ( jr $ra )`
- I can store the value of `$ra` in the stack and can be restored when it is time to return to the caller
- This allows me to call a subroutine from inside a subroutine without losing the return address
- When I call another subroutine,