Steps to Debug Assembly Code in GDB and See Register Values

Information provided by Isabella Steinebach (CMSC216 Summer 2025 TA)

1. Run GDB for your file:
   ○ Ex: make palindrome_s.gdb
2. Set a breakpoint wherever you want to start debugging
   ○ Ex: b is_palindrome
3. Type "layout regs" in the gdb command line.
4. Continue typing "layout next" until you see two tabs with the register group as one window, and your source code (palindrome.S) in the next window.

```
─┌─TERMINAL                                          ⟩_ ssh ╋ ∨ ▯ 🗑 ···
  ┌─Register group: general─────────────────────────────────────────────
  │r0              0x1       1
  │r1              0x0       0
  │r2              0x0       0
  │r3              0x0       0
  │r4              0x0       0
  │r5              0x0       0
  │r6              0x0       0
  │r7              0x0       0
  │r8              0x0       0
    ┌─palindrome.S──────────────────────────────────────────────────
    │1       .text
    │2
    │3       .global is_palindrome
    │4       is_palindrome:
  B+>│5          ret^?
    │6
    │7
    │8
    │9
    └────────────────────────────────────────────────────────────
  remote Thread <main> In: is_palindrome                      L5    PC: 0xea
  (gdb) layout next
  (gdb) layout next
```

Now you can see the content of each register as you are stepping through your source code. This is very helpful to be able to debug your code.

Note: The white border around the palindrome.S window in the above screenshot means I am currently "focused" on that window. This allows me to scroll in that window through my code, but all other windows are frozen. To focus on the register window and scroll through it, run "focus regs" and then you can scroll through the register window. To return the focus back to the source code window, run "focus src".