```
gcd:
        cmp  r0, r1      # compare r0 and r1
        beq  end        # branch if r0 == r1
        blt  less       # branch if r0 < r1
        sub  r0, r0, r1  # r0 = r0 - r1
        b    gcd        # branch to label gcd
less:
        sub  r1, r1, r0  # r1 = r1 - r0
        b    gcd        # branch to label gcd
end:
        add  r1, r1, r0  # r1 = r1 + r0
        mov  r3, r1     # r3 = r1
```

Sample code for GCD with conditional branches

```
add  r0, r1, r2     # r0 = r1 + r2
sub  r1, r0, r1     # r1 = r0 - r1
ldr  r2, [r1]       # r2 = mem[r1]
sub  r0, r2, r1     # r0 = r2 - r1
cmp  r0, r3         # compare r0 and r3
```

Sample code with read-after-write data hazards