```
promptA: DC "Enter first positive integer a:\0"
promptB: DC "Enter second positive integer b:\0"
result:  DC "GCD is:\0"
STACK:   EQU 0x100000

lui sp, STACK>>12          ; initialize stack pointer

addi x5, x0, promptA
ecall x1, x5, 4            ; print prompt for a
ecall a0, x0, 5            ; read a → a0

addi x5, x0, promptB
ecall x1, x5, 4            ; print prompt for b
ecall a1, x0, 5            ; read b → a1

jal x1, gcd                ; call gcd(a0, a1)

addi x5, x0, result
ecall x1, x5, 4            ; print label
ecall x0, a0, 0            ; print gcd result
ebreak x0, x0, 0           ; stop

# ---------------------------
# gcd(a0 = x, a1 = y)
# if (y == 0) return x;
# else gcd(y, x % y);
# ---------------------------
gcd:
beq a1, x0, base           ; if y == 0, return a

sd x1, -8(sp)              ; push return address
sd a0, -16(sp)             ; push a
sd a1, -24(sp)             ; push b
addi sp, sp, -24           ; adjust stack

rem a2, a0, a1             ; a2 = a % b
addi a0, a1, 0             ; a0 = b
addi a1, a2, 0             ; a1 = a % b
jal x1, gcd                ; recursive call

addi sp, sp, 24            ; restore sp
ld x1, -8(sp)              ; pop return address
```

```
jalr x0, 0(x1)          ; return (a0 already has result)

base:
jalr x0, 0(x1)          ; return a0
```