.ORIG x3000

LD R0, NUM

LD R1, DEN

JSR GCD

ADD R4, R1, #0

ADD R1, R2, #0

JSR DIVIDE

ST R2, NUM

ADD R0, R4, #0

JSR DIVIDE

ST R2, DEN

HALT

; you can try other values for NUM and DEN by replacing these values in the simulator

NUM .FILL #200 ; you can try other values for NUM and DEN by replacing

DEN .FILL #30

; Divide R0 by R1, putting quotient in R2 and remainder in R3

DIVIDE ST R0, SR0

ST R1, SR1

AND R2, R2, #0

AND R3, R3, #0

NOT R1, R1

ADD R1, R1, #1

DIVIDEII1 ADD R2, R2, #1

ADD R0, R0, R1

BRzp DIVIDEII1

ADD R2, R2, #-1

ADD R1, R1, #-1

NOT R1, R1

ADD R0, R0, R1

ADD R3, R0, #0

RET

SR0 .BLKW 1

SR1 .BLKW 1

; Euclid's algorithm for GCD of R0 and R1, result in R2

GCD ST R0, SR0A

ST R1, SR1A

ADD R6, R7, #0

GCDL JSR DIVIDE

ADD R0, R1, #0

ADD R1, R3, #0

BRp GCDL

ADD R2, R0, #0

LD R0, SR0A

LD R1, SR1A

JMP R6

SR0A .BLKW 1

SR1A .BLKW 1

.END