

## CSE 3666 - Lab 1

Jonathan Ameri

```
1  #      CSE 3666 Lab 1
2
3      .globl main
4
5      .text
6  main:
7
8      # read two positive integers from the console and
9      # save them in s1 and s2
10     # compute the GCD of the two numbers with Euclidean algorithm
11     #   while a != b:
12     #       if a > b:
13     #           a = a - b
14     #       else:
15     #           b = b - a
16     # print the GCD
17
18     # GCD examples:
19     #   gcd(11, 121) = 11
20     #   gcd(24, 60) = 12
21     #   gcd(192, 270) = 6
22     #   gcd(14, 97) = 1
23
24     # use system call 5 to read integers
25     addi a7, x0, 5
26     ecall
27     addi s1, a0, 0      # a in s1
28
29     # using pseudoinstructions
30     li a7, 5
31     ecall
32
33     mv s2, a0          # b in s2
34     #mv s2, a0 -> s2 = a0
35
36     # TODO
37     # Add you code here
38     # compute GCD(a, b) and print it
39
40     beq x0, x0, test
41 loop: blt s1, s2, case1  #if s1 < s2 goto case1
42     sub s1, s1, s2      #else s1 = s1-s2
43     beq x0, x0, test
44 case1: sub s2, s2, s1    #s2 = s2-s1
45     bne s1, s2, loop    #test if a != b
46
47     li a7, 1
48     add a0, s1, zero
49     ecall
50
51     # sys call to exit
52     addi a7, x0, 10
53     ecall
```

The code works for the majority of inputs. For numbers that have a GCD greater than 1, the program prints that GCD. For numbers with no GCD other than 1, the program prints 1. The program will go into an infinite loop however if one or both of the numbers are negative. This is demonstrated below:

```
192
60
12
-- program is finished running (0) --

66
22
22
-- program is finished running (0) --

45
20
5
-- program is finished running (0) --

192
76
4
-- program is finished running (0) --

11
-4
```