Algorithms: GCD analysis

Review questions

1 What is 27 mod 5?

2 What is 2 mod 5?

3 Which of the following statements is true, assuming that a and b are positive integers?

- $0 \leq a \bmod b < b$
- $0 \leq a \bmod b < a$

4 What is 5 mod 0?

5 Is 0 divisible by 10?

Model 1: GCD

Definition 1. Recall that the *greatest common divisor*, or GCD, of two positive integers a and b is defined as the largest positive integer which evenly divides both a and b. The GCD of a and b is denoted $\gcd(a, b)$.

6 What is $\gcd(12, 30)$?

7 What are the prime factorizations of 12 and 30?

8 What do the prime factorizations of 12 and 30 have to do with $\gcd(12, 30)$?

9 What is $\gcd(144, 690)$?

10 What if we extend the definition of GCD to apply to all nonnegative integers? What should $\gcd(a, 0)$ be when $a > 0$?

Model 2: The Euclidean Algorithm

Consider the four algorithms specified below. They are all supposed to compute the GCD of nonnegative integers, but only two of them are correct.

```
GCDIa(m,n) =
  a ← m
  b ← n
  while (a ≠ 0)
    if a ≤ b
      then b ← b mod a
      else a ← a mod b
  if a = 0 then return b else return a
```

```
GCDIb(m,n) =
  a ← m
  b ← n
  while (a ≠ 0) and (b ≠ 0)
    if a ≤ b
      then b ← b mod a
      else a ← a mod b
  if a = 0 then return b else return a
```

```
GCDRa(a,b) =
  if b = 0
    then a
    else GCDRa(b, a mod b)
```

```
GCDRb(a,b) =
  if b = 0
    then a
    else GCDRb(a mod b, b)
```

11 Trace the execution of each algorithm on the inputs $(144, 690)$.

12 What do you think the I and R stand for in GCDI and GCDR?

13 List some similarities and differences among the algorithms.

14 Which algorithms are incorrect? What is wrong with them?

15 For the correct algorithms, describe in a few sentences what happened to the values of a and b as the algorithm ran. Can you explain why the algorithms will always stop eventually?

16 Look at one of your execution traces from Question 11. Find the gcd of a and b after each iteration of the algorithm. What do you notice?