Math 422 HMWK 4 Selected Solutions

Savage

Exercise A

Theorem 0.1. *Let $a, b \in \mathbb{Z}$ and let d be any positive common divisor of a and b. Then*

$$\gcd(a, b) = d \gcd(a/d, b/d).$$

Proof. Let $g = \gcd(a, b)$ and $g' = \gcd(a/d, b/d)$. We know that there exist integers x and y such that

$$ax + by = g.$$

Dividing both sides by d we get $(a/d)x + (b/d)y = g/d$. One should note each fraction is an integer as any divisor of a and b divides $\gcd(a, b)$. We claim that g/d is the smallest positive integer that can be written as a positive linear combination of a/d and b/d. To see this, if $g/d = 1$, then clearly the statement holds. So assume $g/d > 1$ and suppose by contradiction that we could find integers m, n, and t where $0 < t < g/d$ satisfying

$$(a/d)m + (b/d)n = t.$$

Multiplying both sides by d implies

$$am + bn = td < g.$$

This is a contradiction since g was the smallest positive integer that could be written as a linear combination of a and b. This shows $g/d = \gcd(a/d, b/d)$ and the result follows. □

Exercise C

We begin by noting that a number ends in zero if and only if it is divisible by $10 = 2 \cdot 5$. So the question about finding the number of zeros is equivalent to finding how many pairs of 2's and 5's we can make in 100!. We begin with counting how many 2's we have. Note that since there are 50 distinct even numbers in the product of 100! that $2^{50}|100!$. This means 100!, in its prime factorization, has at least 50 2's. Next, take note that exactly twenty terms of 100! are divisible by 5. Of those 20 terms, four are divisible by 5^2 and none are divisible by 5^k where $k \geq 3$. This shows that exactly 24 5's appear in the prime factorization of 100!. Observe the most number of pairings of 2 and 5 that can be created is 24, so 100! will end in 24 zeros.

Exercise E

Silverman 8.7

We provide the code and prove it works. Note: remember to have your program call your egcd function if it is saved on a different file.

```
def ModSolve(a,c,m):
        """returns all unique solutions x to ax=c mod m"""

        g=egcd(a,m)[0] # gcd(a,m)
        solutions=[]
        if c%g!=0:
                return solutions
        I=egcd(a,m)[1]*c/g # initial solution
        for i in range(g):
                solutions=solutions+[I+i*m/g]
        return solutions
```

To prove this code works, we need to prove three things.

1. The code terminates.
2. The output only contains solutions.
3. The output contains all possible solutions.

Theorem 8.1 is precisely (actually a little stronger) what we need to prove.