MATH 350 ASSIGNMENT 7, FALL 2015

Note: The midterm exam will cover materials up to chapter 23 including congruent number systems $\mathbb{Z}/n\mathbb{Z}$'s, Fermat's little theorem and its generalizations, Euler's ϕ-function, prime numbers, quadratic residue, Legendre symbols and Jacobi symbols, quadratic reciprocity. The emphasis is on the basics.

Due in class on Friday, October 23

Read Chapters 20–23
Part 1. From the textbook *A friendly introduction to number theory*.

- Exercise 22.3, 22.7
- Exercise 23.4.

Part 2. Extra credits.

A. We showed in class that

$$\left(\frac{a}{p}\right) = (-1)^{\Sigma_{k=1}^{(p-1)/2} \lfloor \frac{2ak}{p} \rfloor}$$

for every integer a, where p is any odd prime number. (This equality is not in the book.) We have also seen that if in addition a is odd and not divisible by p, we have

$$\left(\frac{a}{p}\right) = (-1)^{\Sigma_{k=1}^{(p-1)/2} \lfloor \frac{ak}{p} \rfloor}.$$

(This is Lemma 23.3.) Combining the two, we have that the following congruence

$$\sum_{k=1}^{(p-1)/2} \left\lfloor \frac{2ak}{p} \right\rfloor \equiv \sum_{k=1}^{(p-1)/2} \left\lfloor \frac{ak}{p} \right\rfloor \pmod 2$$

whenever p is an odd prime number and a is an odd integer not divisible by p. Can you prove the above congruence property directly?

B. We have said many times in class that the Euclidean algorithm for finding the gcd of two integers is "fast", and that using the Jacobi symbol one can compute the Legendre symbol "fast". Suppose that p is prime number with n binary digits (say around 1,000), and a is an integer not divisible by p, with m binary digits (say also around 1,000). Estimate how many divisions algorithms a computer may need to do in the process of computing the Legendre symbol $\left(\frac{a}{p}\right)$? (Please produce an upper bound.)