

Due: February 12, 2008

1 Least Common Multiple

In class we talked about the Greatest Common Divisor, or GCD. A related concept is the Least Common Multiple, or LCM. For integers a, b , we define $\text{LCM}(a, b)$ to be the *smallest* integer that is a multiple of both a and b .

1.1 Algorithm (15 points)

Write an efficient algorithm that takes in two integers a and b and returns their least common multiple. Your algorithm should be written in clear pseudocode.

1.2 Proof (20 points)

Prove that your code from the previous part computes the LCM correctly and efficiently. If you get stuck, look at the corresponding proof for the Euclidean GCD Algorithm covered in lecture.

2 Sums mod n

2.1 Modulo odds (15 points)

Prove that $\sum_{i=0}^{n-1} i = 0 \pmod{n}$ if $n > 0$ is odd. (Hint: Think about negative numbers mod n .)

2.2 Modulo evens (15 points)

Prove that $\sum_{i=0}^{n-1} i = n/2 \pmod{n}$ if $n > 0$ is even.

3 Factorials mod n

For $n > 1$, define the function:

$$P(n) = \left(\prod_{i=1}^{n-1} i^2 \right) \pmod{n} \quad (1)$$

3.1 Modulo composites (15 points)

Show that if n is not a prime number, then $P(n) = 0$.

3.2 Modulo primes (20 points)

Show that if n is prime, then $P(n) = 1$ (Hint: remember that every nonzero value has an inverse modulo any prime).

4 More factorials mod n (25 points extra credit)

Show that for p prime, $(p-1)! = -1 \pmod{p}$.

Hint: The first part of your proof should argue that 1 and $p-1$ are the only values mod p that are their own inverses. In showing that, the polynomial $x^2 - 1 = 0 \pmod{p}$ is important.