CSE 321 Discrete Structures

Winter 2008

Lecture 9

Number Theory: Modular Arithmetic

Announcements

- Readings
  - Today:
    - Modular Exponentiation
      - 3.5, 3.6 (5th Edition: 2.5)
  - Wednesday:
    - Primality
      - 3.6 (5th Edition: 2.5)
  - Friday:
    - Applications of Number Theory
      - 3.7 (5th Edition: 2.6)

Highlights from Lecture 8

- Modular Arithmetic
  - a mod n: remainder when divided by n
    - $0 \leq a \bmod n \leq n-1$
  - $a \equiv b \pmod n$ means a mod n = b mod n
  - $a +_n b = (a + b) \bmod n$
  - $a \times_n b = (a \times b) \bmod n$
- Finite domain arithmetic
  - Well behaved, especially if n is prime
  - Applications to computing

Hashing

- Map values from a large domain, 0…M-1 in a much smaller domain, 0…n-1
- Index lookup
- Test for equality
- Hash(x) = x mod p
- Often want the hash function to depend on all of the bits of the data
  - Collision management

Pseudo Random number generation

- Linear Congruential method

$$x_{n+1} = (a\, x_n + c) \bmod m$$

Data Permutations

- Caesar cipher, a = 1, b = 2, . . .
  - HELLO WORLD
- Shift cipher
  - $f(x) = (x + k) \bmod n$
  - $f^{-1}(x) = (x - k) \bmod n$
- Affine cipher
  - $f(x) = (ax + b) \bmod n$
  - $f^{-1}(x) = (a^{-1}(x-b)) \bmod n$

a	b	c	d	e	f	g
1	2	3	4	5	6	7
5	6	7	1	2	3	4
5	3	1	6	4	2	7

Modular Exponentiation

x	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	4	6	1	3	5
3	3	6	2	5	1	4
4	4	1	5	2	6	3
5	5	3	1	6	4	2
6	6	5	4	3	2	1

a	a^1	a^2	a^3	a^4	a^5	a^6
1						
2						
3						
4						
5						
6						

Fermat's Little Theorem

- If p is prime, $0 < a \leq p-1$, $a^{p-1} \equiv 1 \pmod{p}$
- Group theory
  - Index of x, smallest $i > 0$ such that $x^i = 1$
  - The index of x divides the order of the group

Exponentiation

- Compute 78365^{81453}
- Compute 78365^{81453} mod 104729

Fast exponentiation

```
int FastExp(int x, int n){
    long v = (long) x;
    int m = 1;
    for (int i = 1; i <= n; i++){
        v = (v * v) % modulus;
        m = m + m;
        Console.WriteLine("i : " + i + ", m : " + m + ", v : " + v  );
    }
    return (int)v;
}
```

Program Trace

```
i : 1, m : 2, v : 82915
i : 2, m : 4, v : 95592
i : 3, m : 8, v : 70252
i : 4, m : 16, v : 26992
i : 5, m : 32, v : 74970
i : 6, m : 64, v : 71358
i : 7, m : 128, v : 20594
i : 8, m : 256, v : 10143
i : 9, m : 512, v : 61355
i : 10, m : 1024, v : 68404
i : 11, m : 2048, v : 4207
i : 12, m : 4096, v : 75698
i : 13, m : 8192, v : 56154
i : 14, m : 16384, v : 83314
i : 15, m : 32768, v : 99519
i : 16, m : 65536, v : 29057
```

Fast exponentiation algorithm

- What if the exponent is not a power of two?

$81453 = 2^{16} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^9 + 2^5 + 2^3 + 2^2 + 2^0$

The fast multiplication algorithm computes a^n mod p in time O(log n)

Discrete Log Problem

- Given integers a, b in [1,…, p-1], find k such that $a^k \bmod p = b$

Primality

- An integer p is prime if its only divisors are 1 and p
- An integer that is greater than 1, and not prime is called composite
- Fundamental theorem of arithmetic:
  - Every positive integer greater than one has a unique prime factorization

Factorization

- If n is composite, it has a factor of size at most sqrt(n)

Euclid's theorem

- There are an infinite number of primes.
- Proof by contradiction:
- Suppose there are a finite number of primes: $p_1, p_2, \ldots p_n$