Discrete Mathematics

CS204: Spring, 2008

Jong C. Park
Computer Science Division, KAIST

Today's Topics

Divisors

Representations of Integers and Integer Algorithms

Introduction to Number Theory

Introduction to Number Theory

- Divisors
- Representations of Integers and Integer Algorithms
- The Euclidean Algorithm
- The RSA Public-Key Cryptosystem

Divisors

- Definition
  - Let n and d be integers, $d \neq 0$. We say that d divides n if there exists an integer q satisfying $n = dq$. We call q the quotient and d a divisor or factor of n. If d divides n, we write $d \mid n$. If d does not divide n, we write $d \nmid n$.
- Example
  - Since $21 = 3 \cdot 7$, 3 divides 21 and we write $3 \mid 21$. The quotient is 7. We call 3 a divisor or factor or 21.
  - Show that if n and d are positive integers and $d \mid n$, then $d \leq n$.

Divisors

- Note
  - Whether an integer $d > 0$ divides an integer n or not, we obtain a unique quotient q and remainder r as given by the Quotient-Remainder Theorem:
    - There exist unique integers q (quotient) and r (remainder) satisfying $n = dq + r$, $0 \leq r < d$.
  - The remainder r equals zero if and only if d divides n.

Divisors

- Theorem
  - Let m, n, and d be integers.
    - (a) If $d \mid m$ and $d \mid n$, then $d \mid (m + n)$.
    - (b) If $d \mid m$ and $d \mid n$, then $d \mid (m - n)$.
    - (c) If $d \mid m$, then $d \mid mn$.
  - Proof.
    - Exercise

Divisors

- Definition
  - An integer greater than 1 whose only positive divisors are itself and 1 is called prime. An integer greater than 1 that is not prime is called composite.
- Examples
  - Show that the integer 23 is prime.
    - 1, 23
  - Show that the integer 34 is composite.
    - 1, 17, 34

Divisors

- Note
  - To determine if a positive integer *n* is composite, it suffices to test whether any of the integers 2, 3, ..., *n* – 1 divides *n*.
  - If some integer in this list divides *n*, then *n* is composite.
  - If no integer in this list divides *n*, then *n* is prime.
- Examples
  - Show that 43 is prime.
  - Show that 451 is composite.
    - 11

Divisors

- Theorem
  - A positive integer n greater than 1 is composite if and only if n has a divisor d satisfying $2 \leq d \leq \sqrt{n}$.
  - Proof.
    - We must prove the following two claims.
      - If n is composite, then n has a divisor d satisfying $2 \leq d \leq \sqrt{n}$.
      - If n has a divisor d satisfying $2 \leq d \leq \sqrt{n}$, then n is composite.

Divisors

Algorithm 5.1.8: Testing Whether an Integer is Prime

Input: n

Output: d

$is\_prime(n)$ {
    for $d = 2$ to $\lfloor \sqrt{n} \rfloor$
        if (n mod $d == 0$)
            return d
    return 0
}

Divisors

- Examples
  - Determine whether 43 is prime, using the earlier algorithm.
    - The algorithm check whether any of 2, 3, 4, 5, 6 = $\lfloor\sqrt{43}\rfloor$ divides 43.
    - None of these numbers divides 43, so the condition n mod d == 0 in the algorithm is always false.
    - Therefore, the algorithm returns 0 to indicate that 43 is prime.
  - Determine whether 451 is prime.

Divisors

- Example
  - If the input the earlier algorithm is $n = 1274$, the algorithm returns the prime 2 because 2 divides 1274, specifically $1274 = 2 \cdot 637$.
  - If we input $n = 637$, we get the prime 7, specifically $637 = 7 \cdot 91$.
  - With $n = 91$, we get the prime 7 again, specifically $91 = 7 \cdot 13$.
  - If we now input $n = 13$, the algorithm returns 0 because 13 is prime.
  - Combining the previous equations, we get $1274 = 2 \cdot 7 \cdot 7 \cdot 13$.

Divisors

- Theorem
  - Fundamental Theorem of Arithmetic
    - Any integer greater than 1 can be written as a product of primes. Moreover, if the primes are written in nondecreasing order, the factorization is unique. In symbols, if

      $$n = p_1 p_2 \cdots p_i,$$

      where the p_k are primes and $p_1 \leq p_2 \leq \cdots \leq p_i$, and

      $$n = p'_1 p'_2 \cdots p'_j,$$

      where the p'_k are primes and $p'_1 \leq p'_2 \leq \cdots \leq p'_j$, then $i = j$ and

      $$p_k = p'_k \text{ for all } k = 1, \ldots, i.$$

Divisors

- Theorem
  - The number of primes is infinite.
    - Proof.
      - It suffices to show that if p is a prime, there is a prime larger than p.
      - To this end, we let $p_1, p_2, ..., p_n$ denote all of the distinct primes less than or equal to p.
      - Consider the integer $m = p_1 p_2 \cdots p_n + 1$.
      - (Complete the proof.)

Divisors

- Definition
  - Let *m* and *n* be integers with not both *m* and *n* zero. A common divisor of *m* and *n* is an integer than divides both *m* and *n*. The greatest common divisor, written gcd(*m*,*n*), is the largest common divisor of *m* and *n*.
- Example
  - What is the greatest common divisor of 30 and 105?
    - We can find the answer by enumerating the positive divisors of each number.
    - We can also find the answer by inspecting the prime factorization of each number.

Divisors

- Theorem
  - Let m and n be integers, $m > 1$, $n > 1$, with prime factorizations
    $$m = p^{a_1}{}_1 p^{a_2}{}_2 \cdots p^{a_n}{}_n$$
    and
    $$n = p^{b_1}{}_1 p^{b_2}{}_2 \cdots p^{b_n}{}_n.$$
    (If the prime p_i is not a factor of m, we let $a_i = 0$. Similarly, if the prime p_i is not a factor of n, we let $b_i = 0$.)
    Then $\gcd(m,n) = p^{\min(a_1,b_1)}{}_1 p^{\min(a_2,b_2)}{}_2 \cdots p^{\min(a_n,b_n)}{}_n$.
- Example
  - What is the greatest common divisor of 82320 and 950796?
    - $\gcd(82320,950796) = 2^{\min(4,2)} \cdot 3^{\min(1,2)} \cdot 5^{\min(1,0)} \cdot 7^{\min(3,4)} \cdot 11^{\min(0,1)}$
      $= 2^2 \cdot 3^1 \cdot 5^0 \cdot 7^3 \cdot 11^0 = 4116.$

Divisors

- Definition
  - Let m and n be positive integers. A common multiple of m and n is an integer that is divisible by both m and n. The least common multiple, written lcm(m,n), is the smallest positive common multiple of m and n.
- Example
  - The least common multiple of 30 and 105
    - Use the “list all divisors” method.
    - Use the prime factorization method.

Divisors

- Theorem
  - Let m and n be integers, $m > 1$, $n > 1$, with prime factorizations

    $$m = p^{a_1}{}_1 p^{a_2}{}_2 \cdots p^{a_n}{}_n$$

    and

    $$n = p^{b_1}{}_1 p^{b_2}{}_2 \cdots p^{b_n}{}_n.$$

    (If the prime p_i is not a factor of m, we let $a_i = 0$. Similarly, if the prime p_i is not a factor of n, we let $b_i = 0$.)

    Then $\text{lcm}(m,n) = p^{\max(a_1,b_1)}{}_1 p^{\max(a_2,b_2)}{}_2 \cdots p^{\max(a_n,b_n)}{}_n$.
- Example
  - What is the least common multiple of 82320 and 950796?

Divisors

- Theorem
  - For any positive integers m and n, $\gcd(m,n) \cdot \mathrm{lcm}(m,n) = mn$.
  - Proof.
    - Exercise
    - Establish the claim first with m = 1, and separately with n = 1, and then assume m > 1 and n > 1.
    - Use the fact that min(x,y) + max(x,y) = x + y.

Representations of Integers and Integer Algorithms

- Terminology
  - bit
  - the binary number system
  - the hexadecimal number system
  - the octal number system
  - the base of the number system
- Example
  - Computer Representation of Integers
    - What is the number of bits required to represent n?
      - $\lfloor 1 + \lg n \rfloor$

Representations of Integers and Integer Algorithms

- Example
  - Binary to Decimal
    - 101101_2
    - 45_{10}.

Representations of Integers and Integer Algorithms

Algorithm 5.2.3: Converting an Integer from Base b to Decimal

$base\_b\_to\_dec(c, n, b)$
    $dec\_val = 0$
    $power = 1$
    for $i = 0$ to n {
        $dec\_val = dec\_val + c_i * power$
        $power = power * b$
    }
    return dec_val
}

Representations of Integers and Integer Algorithms

- Examples
  - Hexadecimal to Decimal
    - $B4F_{16}$
    - 2895_{10}.
  - Decimal to Binary
    - 130_{10}
    - 10000010_2.

Representations of Integers and Integer Algorithms

Algorithm 5.2.7: Converting a Decimal Integer into Base b

Input: m, b

Output: c, n

$dec\_to\_base\_b(m, b, c, n)$ {
  $n = -1$
  while $(m > 0)$ {
    $n = n + 1$
    $c_n = m \bmod b$
    $m = \lfloor m/b \rfloor$
  }
}

Representations of Integers and Integer Algorithms

- Examples
  - Convert the decimal number $m = 11$ to binary.
  - Decimal to Hexadecimal
    - 20385_{10}
    - $4FA1_{16}$.
  - Binary Addition
    - Add the binary numbers 10011011 and 1011011.
    - 11110110

Representations of Integers and Integer Algorithms

Algorithm 5.2.12: Adding Binary Numbers

Input: b, b', n

Output: s

$binary\_addition(b, b', n, s)$ {
  $carry = 0$
  for $i = 0$ to n {
    $s_i = (b_i + b'_i + carry) \bmod 2$
    $carry = \lfloor (b_i + b'_i + carry)/2 \rfloor$
  }
  $s_{n+1} = carry$
}

Representations of Integers and Integer Algorithms

- Example
  - Hexadecimal Addition
    - Add the hexadecimal numbers 84F and 42EA.

Representations of Integers and Integer Algorithms

- Example
  - Compute a^{29} with repeated squaring.
    - $a^{29} = a^1 \cdot a^4 \cdot a^8 \cdot a^{16}$.
    - Initially, *x* is set to *a*, and *n* is set to the value of the exponent, 29.
    - We then compute *n* mod 2. Since this value is 1, we know that $1 = 2^0$ is included in the binary expansion of 29. Therefore a^1 is included in the product. We track the partial product in *Result*; so *Result* is set to a.
    - We then compute the quotient when 29 is divided by 2. The quotient 14 becomes the new value of *n*.
    - We then repeat this process (until n becomes 0).

Representations of Integers and Integer Algorithms

Algorithm 5.2.16: Exponentiation By Repeated Squaring

Input: a, n

Output: a^n

$exp\_via\_repeated\_squaring(a, n)$ {
    $result = 1$
    $x = a$
    while $(n > 0)$ {
        if $(n \bmod 2 == 1)$
            $result = result * x$
        $x = x * x$
        $n = \lfloor n/2 \rfloor$
    }
    return $result$
}

Representations of Integers and Integer Algorithms

- Theorem
  - If a, b, and z are positive integers,
    $$ab \bmod z = [(a \bmod z)(b \bmod z)] \bmod z.$$
  - Proof.
    - Exercise
- Example
  - Show how to compute $572^{29} \bmod 713$.
    - To compute a^{29}, we successively computed a, $a^5 = a \cdot a^4$, $a^{13} = a^5 \cdot a^8$, $a^{29} = a^{13} \cdot a^{16}$.
    - To compute $a^{29} \bmod z$, we successively compute $a \bmod z$, $a^5 \bmod z$, $a^{13} \bmod z$, $a^{29} \bmod z$.

Representations of Integers and Integer Algorithms

Algorithm 5.2.19: Exponentiation Mod z By Repeated Squaring

Input: a, n, z

Output: $a^n \bmod n$

$exp\_mod\_z\_via\_repeated\_squaring(a, n, z)$ {
    $result = 1$
    $x = a \bmod z$
    while $(n > 0)$ {
        if $(n \bmod 2 == 1)$
            $result = (result * x) \bmod z$
        $x = (x * x) \bmod z$
        $n = \lfloor n/2 \rfloor$
    }
    return $result$
}

Summary

- Divisors
- Representations of Integers and Integer Algorithms