Introduction to Number Theory

Dhananjoy Dey

Indian Institute of Information Technology, Lucknow
ddey@iiitl.ac.in

July 20, 2023

Disclaimers

1

All the pictures used in this presentation are taken from freely available websites.

2

If there is a reference on a slide all of the information on that slide is attributable to that source whether quotation marks are used or not.

What is Number Theory?

What is Number Theory?

NT

Number theory is concerned mainly with the study of the properties of the integers

$$\mathbb{Z} = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots, \},$$

particularly the positive integers $\mathbb{Z}^{+}$ or set of natural numbers $\mathbb{N}$

$$= \{1, 2, 3, \ldots\}.$$

Properties of Natural Numbers

Example

For example all positive integers can be classified into a variety of different types:

Properties of Natural Numbers

Example

For example all positive integers can be classified into a variety of different types:

(i) Unit: 1
(ii) Prime numbers: 2, 3, 5, 7, 11, 13, 17, 19, ...
(iii) Composite numbers: 4, 6, 8, 9, 10, 12, 14, 15, ...

Properties of Natural Numbers

Example

For example all positive integers can be classified into a variety of different types:

(i) Unit: 1

(ii) Prime numbers: 2, 3, 5, 7, 11, 13, 17, 19, . . .

(iii) Composite numbers: 4, 6, 8, 9, 10, 12, 14, 15, . . .

(a) Odd: 1, 3, 5, 7, 9, 11, . . .

(b) Even: 2, 4, 6, 8, 10, . . .

Properties of Natural Numbers

Example

The natural numbers have been separated into a variety of different types

- Square: 1, 4, 9, 16, 25, 36, . . .
- Cube: 1, 8, 27, 64, 125, . . .

Properties of Natural Numbers

Example

The natural numbers have been separated into a variety of different types

- Square: 1, 4, 9, 16, 25, 36, . . .
- Cube: 1, 8, 27, 64, 125, . . .
- Fibonacci: 1, 1, 2, 3, 5, 8, 13, 21, . . .

Properties of Natural Numbers

Example

The natural numbers have been separated into a variety of different types

- Square: 1, 4, 9, 16, 25, 36, ...
- Cube: 1, 8, 27, 64, 125, ...
- Fibonacci: 1, 1, 2, 3, 5, 8, 13, 21, ...
- Perfect: 6, 28, 496, 8128, ...

Properties of Natural Numbers

Example

The natural numbers have been separated into a variety of different types

- Square: 1, 4, 9, 16, 25, 36, ...
- Cube: 1, 8, 27, 64, 125, ...
- Fibonacci: 1, 1, 2, 3, 5, 8, 13, 21, ...
- Perfect: 6, 28, 496, 8128, ...
- Triangular: 1, 3, 6, 10, 15, 21, ...

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.
  - Can the sum of two squares be a square?

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.
  - Can the sum of two squares be a square?

    Yes
  - Can the sum of two cubes be a cube? [Fermat's Last Theorem]

    No
  - Are there infinitely many prime numbers?
  - Are there infinitely many primes of the form $1 \mod 4$?

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.
  - Can the sum of two squares be a square?

    Yes
  - Can the sum of two cubes be a cube? [Fermat's Last Theorem]

    No
  - Are there infinitely many prime numbers?
  - Are there infinitely many primes of the form $1 \mod 4$?
  - Are there infinitely many primes of the form $3 \mod 4$?

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.
  - Can the sum of two squares be a square?

    Yes
  - Can the sum of two cubes be a cube? [Fermat's Last Theorem]

    No
  - Are there infinitely many prime numbers?
  - Are there infinitely many primes of the form $1 \mod 4$?
  - Are there infinitely many primes of the form $3 \mod 4$?

    Yes
  - Which numbers are sums of two squares?
  - Whether there are any triangular numbers that are also square numbers

Number Theoretic Questions

- The main goal of number theory is to find interesting and unexpected relationships between different sorts of numbers and to prove that those relations are true.
  - Can the sum of two squares be a square?

    Yes
  - Can the sum of two cubes be a cube? [Fermat's Last Theorem]

    No
  - Are there infinitely many prime numbers?
  - Are there infinitely many primes of the form $1 \mod 4$?
  - Are there infinitely many primes of the form $3 \mod 4$?

    Yes
  - Which numbers are sums of two squares?
  - Whether there are any triangular numbers that are also square numbers 36

Famous Quotations Related to Number Theory

Quotation

The great mathematician **Carl Friedrich Gauss** called this subject '*arithmetic*' and he said:

"*Mathematics is the queen of sciences and arithmetic the queen of mathematics.*"

Famous Quotations Related to Number Theory

Prof G. H. Hardy

In the 1^{st} quotation Prof Hardy is speaking of the famous Indian Mathematician Ramanujan. This is the source of the often made statement that *Ramanujan knew each integer personally*.

Famous Quotations Related to Number Theory

Prof G. H. Hardy

In the 1^{st} quotation Prof Hardy is speaking of the famous Indian Mathematician Ramanujan. This is the source of the often made statement that *Ramanujan knew each integer personally*.

(i) I remember once going to see him when he was lying ill at Putney. I had ridden in taxi cab number 1729 and remarked that number seemed to me rather dull one and that I hoped it was not an unfavorable omen.

Famous Quotations Related to Number Theory

Prof G. H. Hardy

In the 1^{st} quotation Prof Hardy is speaking of the famous Indian Mathematician Ramanujan. This is the source of the often made statement that *Ramanujan knew each integer personally*.

(i) I remember once going to see him when he was lying ill at Putney. I had ridden in taxi cab number 1729 and remarked that number seemed to me rather dull one and that I hoped it was not an unfavorable omen."No", he replied it is a very interesting number; it is the smallest number expressible as the sum of cubes of two integers in two different ways.

Famous Quotations Related to Number Theory

Prof G. H. Hardy

In the 1^{st} quotation Prof Hardy is speaking of the famous Indian Mathematician Ramanujan. This is the source of the often made statement that *Ramanujan knew each integer personally*.

(i) I remember once going to see him when he was lying ill at Putney. I had ridden in taxi cab number 1729 and remarked that number seemed to me rather dull one and that I hoped it was not an unfavorable omen."No", he replied it is a very interesting number; it is the smallest number expressible as the sum of cubes of two integers in two different ways.

(ii) Pure mathematics is on the whole distinctly more useful than applied. For what is useful above all is technique and mathematical technique is taught mainly through pure mathematics

A Mathematician's Apology

- G. H. Hardy wrote it in November 1940[a].

A Mathematician's Apology

- G. H. Hardy wrote it in November 1940[a].
- Number theorists may be justified in rejoicing that there is one science, at any rate, and that their own, whose very remoteness from ordinary human activities should keep it gentle and clean.
- Hardy was especially concerned that number theory not be used in warfare.
- He was so proud and so humble.

A Mathematician's Apology

- G. H. Hardy wrote it in November 1940[a].
- Number theorists may be justified in rejoicing that there is one science, at any rate, and that their own, whose very remoteness from ordinary human activities should keep it gentle and clean.
- Hardy was especially concerned that number theory not be used in warfare.
- He was so proud and so humble.
- Number theory underlies modern cryptography which is what makes secure on-line communication possible.
- Secure communication is of course crucial in war.

[a] A Mathematician's Apology

Motivation

NT

- Key ideas in number theory include divisibility and the primality of integers.
- Representations of integers, including binary and hexadecimal representations, are part of number theory.
- Number theory has long been studied because of the beauty of its ideas, its accessibility, and its wealth of open questions.
- Mathematicians have long considered number theory to be **pure mathematics**, but it has important applications to **computer science** and **cryptography**.

Computational Number Theory

Computational Number Theory := Number Theory $\oplus$ Computation Theory

$\Downarrow$	$\Downarrow$	$\Downarrow$
Primality Testing	Elementary Number Theory	Computability Theory
Integer Factorization	Algebraic Number Theory	Complexity Theory
Discrete Logarithms	Combinatorial Number Theory	Infeasibility Theory
Elliptic Curves	Analytic Number Theory	Computer Algorithms
Conjecture Verification	Arithmetic Algebraic Geometry	Computer Architectures
Theorem Proving	Probabilistic Number Theory	Quantum Computing
	Applied Number Theory	Biological Computing
$\vdots$	$\vdots$	$\vdots$

Outline

1. Divisibility and Modular Arithmetic
2. Integer Representations and Algorithms
3. Primes and Greatest Common Divisors
4. Prime Numbers
5. Primes Generation

The Floor & Ceiling of a Real Number

Definition

1. *The **floor** or the **greatest integer** function is defined as*

$$\lfloor x \rfloor \;=\; max\{n \in \mathbb{Z} \;:\; n \leq x\}$$

2. *The **ceiling** or the **least integer** function is defined as*

$$\lceil x \rceil \;=\; min\{n \in \mathbb{Z} \;:\; n \geq x\}$$

3. *The **nearest integer** function is defined as*

$$\lfloor x \rceil \;= \lfloor x + 1/2 \rfloor$$

Outline

1. Divisibility and Modular Arithmetic
2. Integer Representations and Algorithms
3. Primes and Greatest Common Divisors
4. Prime Numbers
5. Primes Generation

Division

Definition

If a & b are integers with $a \neq 0$, then a ***divides*** *b if $\exists$ an integer c s/t $b = ac$.*

- When a divides b we say that a is a **factor** or **divisor** of b and that b is a **multiple** of a.
- The notation $a \mid b$ denotes that a divides b.
- If $a \mid b$, then $\frac{b}{a}$ is an integer.
- If a does not divide b, we write $a \nmid b$.

Properties of Divisibility

Theorem

Let $a, b, \ \& \ c$ *be integers, where* $a \neq 0$.

1. (i) *If* $a \mid b$ *and* $a \mid c$, *then* $a \mid (b + c)$;
2. (ii) *If* $a \mid b$, *then* $a \mid bc$ *for all integers* c;
3. (iii) *If* $a \mid b$ *and* $b \mid c$, *then* $a \mid c$.

Corollary

If $a, b, \ \& \ c$ *are integers, where* $a \neq 0$, *s/t* $a \mid b$ *and* $a \mid c$, *then*

$$a \mid (mb + nc)$$

whenever $m \ \& \ n$ *are integers.*

Division Algorithm

- When an integer is divided by a positive integer, there is a **quotient** and a **remainder**. This is traditionally called the “Division Algorithm", but is really a theorem.

Theorem

If $a, d \in \mathbb{Z}$ & $d > 0$, *then* $\exists\,!\ q$ & $r \in \mathbb{Z}$ *s/t*

$$a = q.d + r,\ where\ 0 \leq r < d.$$

d *is called the* ***divisor***, a *is called the* ***dividend***, q *is called the* ***quotient*** *and* r *is called the* ***remainder***.

- We define **div** and **mod** as
  $q = a\ div\ d$ and $r \equiv a \mod d$

Congruence Relation

Definition

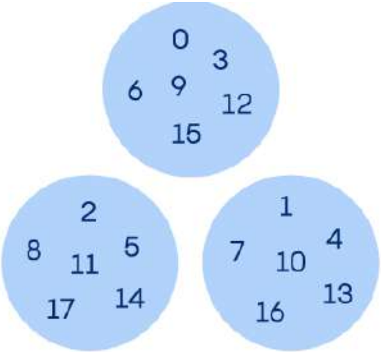
If $a, b \in \mathbb{Z}$ and m is a positive integer, then a is ***congruent to*** *b modulo m if $m \mid (a - b)$.*

- *The notation $a \equiv b \mod m$ says that a is congruent to b modulo m.*
- *We say that $a \equiv b \mod m$ is a* ***congruence*** *and that m is its* ***modulus****.*
- *Two integers are congruent $\mod m$ iff they have the same remainder when divided by m.*
- *If a is not congruent to b modulo m, we write*

$$a \not\equiv b \mod m$$

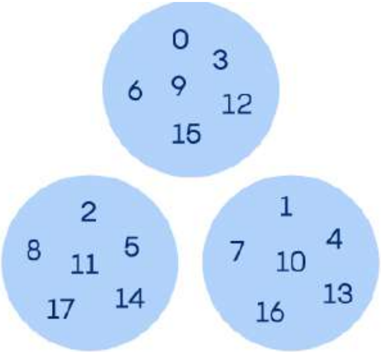
Congruence Relation

Example



Congruence Relation

Example

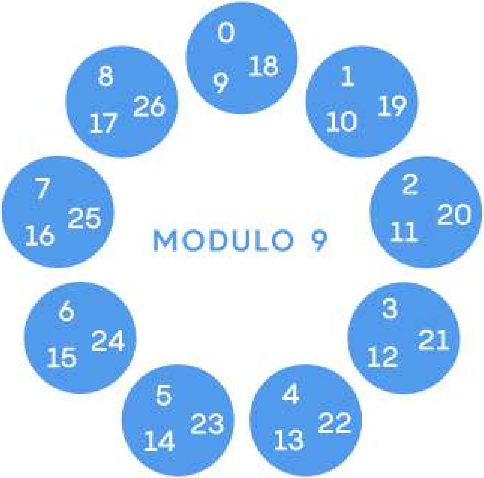


Exercise

Find the modulus.

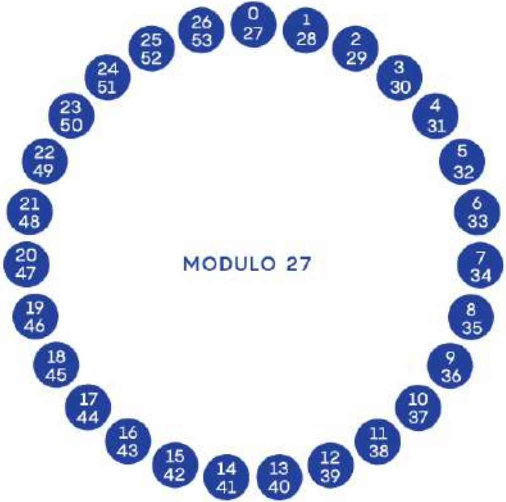
Congruence Relation

Example



Congruence Relation

Example



Congruence Relation

Theorem

Let m be a positive integer. The integers a & b are congruent modulo m iff there is an integer k s/t $a = b + km$.

Congruence Relation

Theorem

Let m be a positive integer. The integers a & b are congruent modulo m iff there is an integer k s/t $a = b + km$.

Proof.

- If $a \equiv b \mod m$, then (by the definition) we have $m \mid (a - b)$. Hence, there is an integer k s/t $a - b = km$ and equivalently $a = b + km$.
- Conversely, if there is an integer k s/t $a = b + km$, then $km = a - b$. Hence, $m \mid (a - b)$ and $a \equiv b \mod m$.

□

Congruence Relation

- The use of mod in $a \equiv b \mod m$ and $a \mod m = b$ are *different*.
  - $a \equiv b \mod m$ is a relation on the set of integers.
  - In $a \mod m = b$, the notation $\mod$ denotes a function.
- The relationship between these notations is made clear in the following theorem.

Theorem

Let a & b be integers, and let m be a positive integer. Then

$$a \equiv b \mod m$$

iff

$$a \mod m = b \mod m.$$

Congruences of Sums and Products

Theorem

Let m be a positive integer. If $a \equiv b \mod m$ and $c \equiv d \mod m$, then

$$(a+c) \equiv (b+d) \mod m \text{ and } ac \equiv bd \mod m$$

Congruences of Sums and Products

Theorem

Let m be a positive integer. If $a \equiv b \mod m$ and $c \equiv d \mod m$, then

$$(a+c) \equiv (b+d) \mod m \text{ and } ac \equiv bd \mod m$$

Proof.

- $\because a \equiv b \mod m$ and $c \equiv d \mod m$, there are integers s & t with $b = a + sm$ and $d = c + tm$.
- Therefore,
  - $b + d = (a + sm) + (c + tm) = (a + c) + m(s + t)$ and
  - $bd = (a + sm)(c + tm) = ac + m(at + cs + stm)$.
- Hence, $(a + c) \equiv (b + d) \mod m$ and $ac \equiv bd \mod m$.

□

Algebraic Manipulation of Congruences

- Multiplying both sides of a valid congruence by an integer preserves validity.

  If $a \equiv b \mod m$ holds then $c.a \equiv c.b \mod m$, where c is any integer.

Algebraic Manipulation of Congruences

- Multiplying both sides of a valid congruence by an integer preserves validity.

  If $a \equiv b \mod m$ holds then $c.a \equiv c.b \mod m$, where c is any integer.

- Adding an integer to both sides of a valid congruence preserves validity.

  If $a \equiv b \mod m$ holds then $(c + a) \equiv (c + b) \mod m$, where c is any integer.

Algebraic Manipulation of Congruences

- Multiplying both sides of a valid congruence by an integer preserves validity.

  If $a \equiv b \mod m$ holds then $c.a \equiv c.b \mod m$, where c is any integer.

- Adding an integer to both sides of a valid congruence preserves validity.

  If $a \equiv b \mod m$ holds then $(c + a) \equiv (c + b) \mod m$, where c is any integer.

- Dividing a congruence by an integer does not always produce a valid congruence.

Algebraic Manipulation of Congruences

- Multiplying both sides of a valid congruence by an integer preserves validity.

  If $a \equiv b \mod m$ holds then $c.a \equiv c.b \mod m$, where c is any integer.

- Adding an integer to both sides of a valid congruence preserves validity.

  If $a \equiv b \mod m$ holds then $(c + a) \equiv (c + b) \mod m$, where c is any integer.

- Dividing a congruence by an integer does not always produce a valid congruence.
  E.g., $6 \equiv 15 \mod 9$; however, $\frac{6}{3} \not\equiv \frac{15}{3} \mod 9$

Computing the $\bmod m$ Function of Products and Sums

Corollary

Let m be a positive integer and let a & b be integers. Then

$$(a+b) \bmod m = ((a \bmod m) + (b \bmod m)) \bmod m$$

and

$$ab \bmod m = ((a \bmod m)(b \bmod m)) \bmod m.$$

- Let $\mathbb{Z}_m = \{0, 1, \ldots, m-1\}$
- The operation $+_m$ is defined as $a +_m b = (a+b) \bmod m$.
- The operation $._m$ is defined as $a._m b = (a.b) \bmod m$.
- $(\mathbb{Z}_m, +_m, \cdot_m)$ forms a commutative ring for any $m \in \mathbb{Z}$ and $m > 0$
- $(\mathbb{Z}_p, +_p, \cdot_p)$ forms a field for any prime p

Outline

1. Divisibility and Modular Arithmetic
2. Integer Representations and Algorithms
3. Primes and Greatest Common Divisors
4. Prime Numbers
5. Primes Generation

Representations of a Number

- $(1234)_{10} =$

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} =$

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary
- $(1234)_{10} =$

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary
- $(1234)_{10} = (2322)_8 = 2.8^3 + 3.8^2 + 2.8^1 + 2$ to the base 8 – octal

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary
- $(1234)_{10} = (2322)_8 = 2.8^3 + 3.8^2 + 2.8^1 + 2$ to the base 8 – octal
- $(1234)_{10} =$

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary
- $(1234)_{10} = (2322)_8 = 2.8^3 + 3.8^2 + 2.8^1 + 2$ to the base 8 – octal
- $(1234)_{10} = (4D2)_{16} = 4.16^2 + D.16^1 + 2.16^0$ to the base 16 – hexadecimal

Representations of a Number

- $(1234)_{10} = 1.10^3 + 2.10^2 + 3.10^1 + 4.10^0$ to the base 10 – decimal
- $(1234)_{10} = (10011010010)_2$

  $1.2^{10} + 0.2^9 + 0.2^8 + 1.2^7 + 1.2^6 + 0.2^5 + 1.2^4 + 0.2^3 + 0.2^2 + 1.2^1 + 0.2^0$ to the base 2 – binary
- $(1234)_{10} = (2322)_8 = 2.8^3 + 3.8^2 + 2.8^1 + 2$ to the base 8 – octal
- $(1234)_{10} = (4D2)_{16} = 4.16^2 + D.16^1 + 2.16^0$ to the base 16 – hexadecimal
- $(BAD)_{26} = (679)_{10} = B.26^2 + A.26 + 26^0$

Revisit

- *Computational complexity theory*

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    (i) Is a problem **P** intrinsically "*easy*" or "*difficult*" to solve?

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    - (i) Is a problem $\mathbf{P}$ intrinsically "*easy*" or "*difficult*" to solve?
    - (ii) Given two problems, $\mathbf{P_1}$ and $\mathbf{P_2}$, which is easier to solve?

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    - (i) Is a problem $\mathbf{P}$ intrinsically "*easy*" or "*difficult*" to solve?
    - (ii) Given two problems, $\mathbf{P_1}$ and $\mathbf{P_2}$, which is easier to solve?
- Running time -

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    - (i) Is a problem $\mathbf{P}$ intrinsically "*easy*" or "*difficult*" to solve?
    - (ii) Given two problems, $\mathbf{P_1}$ and $\mathbf{P_2}$, which is easier to solve?
- Running time - the number of basic (or primitive) operations (or steps) taken by an algorithm.
  - The running time of an algorithm usually depends on

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    (i) Is a problem $\mathbf{P}$ intrinsically "*easy*" or "*difficult*" to solve?
    (ii) Given two problems, $\mathbf{P_1}$ and $\mathbf{P_2}$, which is easier to solve?
- Running time - the number of basic (or primitive) operations (or steps) taken by an algorithm.
  - The running time of an algorithm usually depends on **the size of the input**.

Revisit

- *Computational complexity theory* is the study of the minimal resources needed to solve computational problems.
  - Two fundamental questions:
    1. (i) Is a problem $\mathbf{P}$ intrinsically "*easy*" or "*difficult*" to solve?
    2. (ii) Given two problems, $\mathbf{P_1}$ and $\mathbf{P_2}$, which is easier to solve?
- Running time - the number of basic (or primitive) operations (or steps) taken by an algorithm.
  - The running time of an algorithm usually depends on **the size of the input**.
- Space complexity - to measure the amount of temporary storage used when performing a computational task.

Base b Representations

- We can use positive integer b greater than 1 as a base to represent any number

Base b Representations

- We can use positive integer b greater than 1 as a base to represent any number

Theorem

Let $b, n \in \mathbb{Z}$ and $b > 1,\ \&\ n > 0$. Then n can be expressed uniquely as:

$$n = a_k b^k + a_{k-1} b^{k-1} + \ldots + a_1 b + a_0$$

where $k \in \mathbb{Z}, k \geq 0\ \&\ a_0, a_1, \ldots, a_k$ are nonnegative integers $< b$, and $a_k \neq 0$. The $a_j,\ j = 0, \ldots, k$ are called the base-b digits of the representation.

Base b Representations

- We can use positive integer b greater than 1 as a base to represent any number

Theorem

Let $b, n \in \mathbb{Z}$ and $b > 1,\ \&\ n > 0$. Then n can be expressed uniquely as:

$$n = a_k b^k + a_{k-1} b^{k-1} + \ldots + a_1 b + a_0$$

where $k \in \mathbb{Z}, k \geq 0\ \&\ a_0, a_1, \ldots, a_k$ are nonnegative integers $< b$, and $a_k \neq 0$. The $a_j,\ j = 0, \ldots, k$ are called the base-b digits of the representation.

- The representation of n is called the base b expansion of n and is denoted by $(a_k a_{k-1} \ldots a_1 a_0)_b$.

Representation of a Number

- **Numbers in different bases**

Representation of a Number

- **Numbers in different bases**
  Any number n, $b^{k-1} \leq n < b^k$ is a k-digit number to the base b.

Representation of a Number

- **Numbers in different bases**
  Any number n, $b^{k-1} \leq n < b^k$ is a k-digit number to the base b.
- **Number of digits**

Representation of a Number

- **Numbers in different bases**
  Any number n, $b^{k-1} \leq n < b^k$ is a k-digit number to the base b.
- **Number of digits**

$$= [\log_b \ n] + 1.$$

Representation of a Number

- **Numbers in different bases**
  Any number n, $b^{k-1} \leq n < b^k$ is a k-digit number to the base b.
- **Number of digits**

$$= [\log_b \ n] + 1.$$

- **Number of bits**

Representation of a Number

- **Numbers in different bases**
  Any number n, $b^{k-1} \leq n < b^k$ is a k-digit number to the base b.
- **Number of digits**

$$= [\log_b \ n] + 1.$$

- **Number of bits**

$$= [\log_2 \ n] + 1 \approx [1.44 \times \ ln \ n] + 1.$$

Size of Some Mathematical Objects

Example

1. If $\mathbf{A} = \left[\mathbf{a_{ij}}\right]_{\mathbf{r\times s}}$ is a matrix with r rows, s columns, where $\mathbf{a_{ij}} \in \mathbb{Z}_n$, then the size of $\mathbf{A}$

Size of Some Mathematical Objects

Example

1. If $\mathbf{A} = \left[\mathbf{a_{ij}}\right]_{\mathbf{r\times s}}$ is a matrix with r rows, s columns, where $\mathbf{a_{ij}} \in \mathbb{Z}_n$, then the size of $\mathbf{A}$

$$= rs\,(1 + [\log_2\ n])\ bits.$$

Size of Some Mathematical Objects

Example

1. If $\mathbf{A} = \left[\mathbf{a_{ij}}\right]_{\mathbf{r\times s}}$ is a matrix with r rows, s columns, where $\mathbf{a_{ij}} \in \mathbb{Z}_n$, then the size of $\mathbf{A}$

$$= rs\,(1 + [\log_2\ n])\ bits.$$

2. If f is a polynomial of degree d, over $\mathbb{Z}_n$, then the size of f

Size of Some Mathematical Objects

Example

1. If $\mathbf{A} = \left[\mathbf{a_{ij}}\right]_{\mathbf{r \times s}}$ is a matrix with r rows, s columns, where $\mathbf{a_{ij}} \in \mathbb{Z}_n$, then the size of $\mathbf{A}$

$$= rs\,(1 + [\log_2\ n])\ bits.$$

2. If f is a polynomial of degree d, over $\mathbb{Z}_n$, then the size of f

$$= (d+1)\,(1 + [\log_2\ n])\ bits.$$

Algorithm: Constructing Base b Expansions

Result: $(a_{k-1} \dots a_1 a_0)_b$ is base b expansion of n
procedure base b expansion;
$q := n$;
$k := 0$;
while $q \neq 0$ **do**
  $a_k := q \mod b$;
  $q \leftarrow q \; div \; b$;
  $k \leftarrow k + 1$
end
return $(a_{k-1} \dots a_1 a_0)$

Algorithm 1: Base Conversion

Number of Steps for Doing Arithmetic

Number of steps required to add 2 integers a & b

Number of Steps for Doing Arithmetic

Number of steps required to add 2 integers a & b

Input: integers $a \geq b \geq 0$

Output: $a + b$

Algorithm:

while ($b \neq 0$){
$a = a + +$
$b = b - -$
}
output a

Number of Steps for Doing Arithmetic

Number of steps required to add 2 integers a & b

Input: integers $a \geq b \geq 0$

Output: $a + b$

Algorithm:

while ($b \neq 0$){

$a = a + +$

$b = b - -$

}

output a

Number of operations

Number of Steps for Doing Arithmetic

Number of steps required to add 2 integers a & b

Input: integers $a \geq b \geq 0$

Output: $a + b$

Algorithm:

while $(b \neq 0)\{$

$a = a + +$

$b = b - -$

$\}$

output a

Number of operations $= 3b + 1$

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

i. Look at the top and bottom bit and also at whether there's a carry above the top bit.
ii. If both bits are 0 and there is no carry, then put down 0.

Time$(n + m) = k$-bit operations.

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

1. Look at the top and bottom bit and also at whether there's a carry above the top bit.
2. If both bits are 0 and there is no carry, then put down 0.
3. If either both bits are 0 and there is a carry; or one of the bits is 0, the other is 1 and there is no carry, then put down 1.

Time$(n + m) = k$-bit operations.

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

i. Look at the top and bottom bit and also at whether there's a carry above the top bit.
ii. If both bits are 0 and there is no carry, then put down 0.
iii. If either both bits are 0 and there is a carry; or one of the bits is 0, the other is 1 and there is no carry, then put down 1.
iv. If either one of the bits is 0, the other is 1, and there is a carry; or both bits are 1 and there is no carry then put down 0, put a carry in the next column.

Time$(n+m) = k$-bit operations.

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

i. Look at the top and bottom bit and also at whether there's a carry above the top bit.
ii. If both bits are 0 and there is no carry, then put down 0.
iii. If either both bits are 0 and there is a carry; or one of the bits is 0, the other is 1 and there is no carry, then put down 1.
iv. If either one of the bits is 0, the other is 1, and there is a carry; or both bits are 1 and there is no carry then put down 0, put a carry in the next column.
v. If both bits are 1 and there is a carry, then put down 1, put a carry in the next column.

Bit Operation for Doing Arithmetic

Number of bit operations required to add 2 k-bit integers n & m

i. Look at the top and bottom bit and also at whether there's a carry above the top bit.
ii. If both bits are 0 and there is no carry, then put down 0.
iii. If either both bits are 0 and there is a carry; or one of the bits is 0, the other is 1 and there is no carry, then put down 1.
iv. If either one of the bits is 0, the other is 1, and there is a carry; or both bits are 1 and there is no carry then put down 0, put a carry in the next column.
v. If both bits are 1 and there is a carry, then put down 1, put a carry in the next column.

$$\text{Time}(n+m) = k\text{-bit operations.}$$

Algorithm: Addition of Integers

Number of bit operations required to add 2 k-bit integers n & m

Input: $n = n_k n_{k-1} \cdots n_2 n_1$ & $m = m_k m_{k-1} \cdots m_2 m_1$

Output: $n + m$ in binary.

Algorithm: $c \leftarrow 0$

for($i = 1$ to k){
  if $sum(n_i, m_i, c) = 1$ or 3
    then $d_i \leftarrow 1$
    else $d_i \leftarrow 0$

  if $sum(n_i, m_i, c) \geq 2$
    then $c \leftarrow 1$
    else $c \leftarrow 0$}

if $c = 1$ then output $1 d_k d_{k-1} \cdots d_2 d_1$

  else output $d_k d_{k-1} \cdots d_2 d_1$.

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply a k-bit integer n by an ℓ-bit integer m

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply a k-bit integer n by an ℓ-bit integer m

  i. at most ℓ rows can be obtained
  ii. each row consists of a copy of n shifted to the left a certain distance
  iii. suppose there are $\ell' \leq \ell$ rows.
  iv. multiplication task can be broken down into $\ell' - 1$ additions
  v. moving down from the 2^{nd} row to the ℓ'^{th} row, adding each new row to the partial sum of all of the earlier rows
  vi. each addition takes at most k-bit operations
  vii. total number of bit operations is at most $\ell \times k$.

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply a k-bit integer n by an ℓ-bit integer m

  i. at most ℓ rows can be obtained
  ii. each row consists of a copy of n shifted to the left a certain distance
  iii. suppose there are $\ell' \leq \ell$ rows.
  iv. multiplication task can be broken down into $\ell' - 1$ additions
  v. moving down from the 2^{nd} row to the ℓ'^{th} row, adding each new row to the partial sum of all of the earlier rows
  vi. each addition takes at most k-bit operations
  vii. total number of bit operations is at most $\ell \times k$.

Time$(n \times m) < k\ell$-bit operations.

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply two n-bit integers x & y

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply two n-bit integers x & y
- Let $n = 2t$. Then

$$x = 2^t x_1 + x_0 \ \& \ y = 2^t y_1 + y_0$$

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply two n-bit integers x & y
- Let $n = 2t$. Then

$$x = 2^t x_1 + x_0 \text{ \& } y = 2^t y_1 + y_0$$

-

$$x.y = u_2.2^{2t} + u_1.2^t + u_0$$

Bit Operation for Doing Arithmetic

- Number of bit operations required to multiply two n-bit integers x & y
- Let $n = 2t$. Then

$$x = 2^t x_1 + x_0 \text{ \& } y = 2^t y_1 + y_0$$

- $$x.y = u_2.2^{2t} + u_1.2^t + u_0$$

  where $u_0 = x_0.y_0,\ u_2 = x_1.y_1$ & $u_1 = (x_0 + x_1).(y_0 + y_1) - u_0 - u_2$.

Bit Operation for Modular Exponentiation

Exercise

Compute $3^{37} \mod 53$

Bit Operation for Modular Exponentiation

Exercise

Compute $3^{37} \mod 53$

Solution

- *Binary representation of* $37 = 32 + 4 + 1 = 100101$

Bit Operation for Modular Exponentiation

Exercise

Compute $3^{37} \mod 53$

Solution

- *Binary representation of* $37 = 32 + 4 + 1 = 100101$
- *First we repeatedly square* $3 \mod 53$ *until we have worked out* 3^{2^k} *for every* k *s/t* $2^k \leq 37$.
- *We get*
  $3^2 = 9, 3^4 = 9^2 = 81 \equiv 28, 3^8 \equiv 28^2 =$

Bit Operation for Modular Exponentiation

Exercise

Compute $3^{37} \mod 53$

Solution

- *Binary representation of* $37 = 32 + 4 + 1 = 100101$
- *First we repeatedly square* $3 \mod 53$ *until we have worked out* 3^{2^k} *for every* k *s/t* $2^k \leq 37$.
- *We get*
  $3^2 = 9, 3^4 = 9^2 = 81 \equiv 28, 3^8 \equiv 28^2 = 784 \equiv -11(\because 15 \times 53 = 795),$
  $3^{16} \equiv 121 \equiv 15\ , 3^{32} \equiv 225 \equiv 13.$
- *Therefore,*
  $3^{37} \equiv 13 \times 28 \times 3 = 13 \times 84 \equiv 13 \times 31 = 403 \equiv 32.$

Bit Operation for Modular Exponentiation

- Find $b^n \mod m$ efficiently, where $b, n, \ \& \ m$ are large integers.

Bit Operation for Modular Exponentiation

- Find $b^n \mod m$ efficiently, where $b, n, \ \& \ m$ are large integers.
- We use the binary expansion of $n = (a_{k-1}, \ldots, a_1, a_0)_2$, to compute b^n.

Bit Operation for Modular Exponentiation

- Find $b^n \mod m$ efficiently, where $b, n, \ \& \ m$ are large integers.
- We use the binary expansion of $n = (a_{k-1}, \ldots, a_1, a_0)_2$, to compute b^n.

$$b^n = (b)^{a_{k-1}2^{k-1}+\cdots+a_1 2+a_0} = (b)^{a_{k-1}.2^{k-1}} \ldots (b)^{a_1.2} . (b)^{a_0}$$

Bit Operation for Modular Exponentiation

- Find $b^n \mod m$ efficiently, where $b, n, \ \& \ m$ are large integers.
- We use the binary expansion of $n = (a_{k-1}, \ldots, a_1, a_0)_2$, to compute b^n.

$$b^n = (b)^{a_{k-1}2^{k-1}+\cdots+a_1 2+a_0} = (b)^{a_{k-1}.2^{k-1}} \ldots (b)^{a_1.2} . (b)^{a_0}$$

- Therefore, to compute b^n, we need only compute the values of

$$b, b^2, \left(b^2\right)^2 = b^4, \left(b^4\right)^2 = b^8, \ldots, (b)^{2^{k-1}}$$

and the multiply the terms b^{2^j} in this list, where $a_j = 1$.

Bit Operation for Modular Exponentiation

procedure modular exponentiation $b^n \mod m$;
$x := 1$;
$power := b \mod m$;
for $i := 0$ *to* $k - 1$ **do**
  if $a_i = 1$ **then**
    $x \leftarrow (x.power) \mod m$
  end
  $power \leftarrow (power.power) \mod m$
end
return $x \quad \{x \equiv b^n \mod m\}$

Algorithm 2: Modular Exponentiation

Bit Operation for Modular Exponentiation

procedure modular exponentiation $b^n \mod m$;
$x := 1$;
$power := b \mod m$;
for $i := 0$ *to* $k - 1$ **do**
  if $a_i = 1$ **then**
    $x \leftarrow (x.power) \mod m$
  end
  $power \leftarrow (power.power) \mod m$
end
return $x \quad \{x \equiv b^n \mod m\}$

Algorithm 3: Modular Exponentiation

Computational Complexity to compute

Bit Operation for Modular Exponentiation

```
procedure modular exponentiation b^n mod m;
x := 1;
power := b mod m;
for i := 0 to k − 1 do
    if a_i = 1 then
        x ← (x.power) mod m
    end
    power ← (power.power) mod m
end
return x   {x ≡ b^n mod m}
```

Algorithm 4: Modular Exponentiation

Computational Complexity to compute $b^n \mod m = O((\log m)^2 \log n)$

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

i. At the $(j-1)^{th}$ step $(j = 2, 3, \cdots, n-1)$, you are multiplying $j!$ by $j+1$.

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

i. At the $(j-1)^{th}$ step $(j = 2, 3, \cdots, n-1)$, you are multiplying $j!$ by $j+1$.

ii. $n-2$ steps requires to compute $n!$, where each step involves multiplying a partial product by the next integer.

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

i. At the $(j-1)^{th}$ step $(j = 2, 3, \cdots, n-1)$, you are multiplying $j!$ by $j+1$.

ii. $n-2$ steps requires to compute $n!$, where each step involves multiplying a partial product by the next integer.

iii. Product of n k-bit integers will have at most nk bits.

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

i. At the $(j-1)^{th}$ step $(j = 2, 3, \cdots, n-1)$, you are multiplying $j!$ by $j+1$.
ii. $n-2$ steps requires to compute $n!$, where each step involves multiplying a partial product by the next integer.
iii. Product of n k-bit integers will have at most nk bits.
iv. At each step, we require multiplication of an integer with at most k bits by an integer with at most nk bits.
v. The total number of bit operations is bounded by $(n-2)nk^2$.

Bit Operation for Doing Arithmetic

Example

An upper bound for the number of bit operations required to compute $n!$.

1. At the $(j-1)^{th}$ step $(j = 2, 3, \cdots, n-1)$, you are multiplying $j!$ by $j+1$.
2. $n-2$ steps requires to compute $n!$, where each step involves multiplying a partial product by the next integer.
3. Product of n k-bit integers will have at most nk bits.
4. At each step, we require multiplication of an integer with at most k bits by an integer with at most nk bits.
5. The total number of bit operations is bounded by $(n-2)nk^2$.

Time(to compute $n!$) $\leq n^2(ln\ n)^2$.

Big-O

Big-$\mathcal{O}$

Definition

Let $f,\ g : \mathbb{N} \to \mathbb{R}$, $g(x) > 0\ \forall\ x \geq a$, where $a \in \mathbb{N}$. Then $f = \mathcal{O}(g)$ means that $\frac{f(x)}{g(x)}$ is bounded $\forall\ x \geq a$, i.e., $\exists$ a constant $M > 0$ such that

$$|f(x)| \leq M.g(x) \quad \forall\ x \geq a.$$

Big-$\mathcal{O}$

Definition

Let $f, \; g : \mathbb{N} \to \mathbb{R}$, $g(x) > 0 \; \forall \; x \geq a$, where $a \in \mathbb{N}$. Then $f = \mathcal{O}(g)$ means that $\frac{f(x)}{g(x)}$ is bounded $\forall \; x \geq a$, i.e., $\exists$ a constant $M > 0$ such that

$$|f(x)| \leq M.g(x) \quad \forall \; x \geq a.$$

Example

Let $f(n) = 2.n^3 + 3.n^2 + 4.n + 5$ & $g(n) = n^3$.

Big-$\mathcal{O}$

Definition

Let $f,\ g : \mathbb{N} \to \mathbb{R}$, $g(x) > 0\ \forall\ x \geq a$, where $a \in \mathbb{N}$. Then $f = \mathcal{O}(g)$ means that $\frac{f(x)}{g(x)}$ is bounded $\forall\ x \geq a$, i.e., $\exists$ a constant $M > 0$ such that

$$|f(x)| \leq M.g(x) \quad \forall\ x \geq a.$$

Example

Let $f(n) = 2.n^3 + 3.n^2 + 4.n + 5$ & $g(n) = n^3$.

Then $f = \mathcal{O}(g)$, for take $a = 5,\ M = 3$.

The notation Big $\mathcal{O}$ represents an upper bound of the computational complexity of an algorithm in the worst-case scenario.

Big-O

- g is simpler function than f and it does not increase much faster than f.

Big-O

- g is simpler function than f and it does not increase much faster than f.

Example

1. $n^2 = O(n^3 + n^2 ln\ n + 595)$
2. $n^2 = O(e^{n^2})$
3. $e^{-n} = O(n^2)$

Big-$\mathcal{O}$

- g is simpler function than f and it does not increase much faster than f.

Example

1. $n^2 = \mathcal{O}(n^3 + n^2 ln\ n + 595)$
2. $n^2 = \mathcal{O}(e^{n^2})$
3. $e^{-n} = \mathcal{O}(n^2)$
4. $f(n)(= a_0 + a_1 n + \ldots + a_d n^d) = \mathcal{O}(n^d)$

Big-$\mathcal{O}$

- g is simpler function than f and it does not increase much faster than f.

Example

1. $n^2 \;=\; \mathcal{O}(n^3 + n^2 ln\ n + 595)$
2. $n^2 \;=\; \mathcal{O}(e^{n^2})$
3. $e^{-n} \;=\; \mathcal{O}(n^2)$
4. $f(n)(= a_0 + a_1 n + \ldots + a_d n^d) \;=\; \mathcal{O}(n^d)$
5. $ln\ n \;=\; \mathcal{O}(n^\delta)\ for\ any\ \delta \in \mathbb{R}^+$

Small-o

Small-o

Definition

Let f and g be 2 +ve real valued functions such that

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} \to 0.$$

Then we say that $f = o(g), \Rightarrow f(n) \ll g(n)$ when n is large.

- A function f is **negligible** if $f = o(1/g)$ for any polynomial $g(n) = n^c$

Small-o

Definition

Let f and g be 2 +ve real valued functions such that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \to 0.$$

Then we say that $f = o(g), \Rightarrow f(n) \ll g(n)$ when n is large.

- A function f is **negligible** if $f = o(1/g)$ for any polynomial $g(n) = n^c$
- The notation $g = \Omega(f)$ means exactly the same thing as $f = O(g)$.

Small-o

Definition

Let f and g be 2 +ve real valued functions such that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \to 0.$$

Then we say that $f = o(g), \Rightarrow f(n) \ll g(n)$ when n is large.

- A function f is **negligible** if $f = o(1/g)$ for any polynomial $g(n) = n^c$
- The notation $g = \Omega(f)$ means exactly the same thing as $f = \mathcal{O}(g)$.
- If $f = \mathcal{O}(g)$ and $f = \Omega(g)$ then we use the notation $f = \Theta(g) \;\Rightarrow C_1.g(n) \leq f(n) \leq C_2.g(n)$ for $n \geq n_0, C_i \in \mathbb{R}^+$.

Small-o

Definition

Let f and g be 2 +ve real valued functions such that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} \to 0.$$

Then we say that $f = o(g), \Rightarrow f(n) \ll g(n)$ when n is large.

- A function f is **negligible** if $f = o(1/g)$ for any polynomial $g(n) = n^c$
- The notation $g = \Omega(f)$ means exactly the same thing as $f = \mathcal{O}(g)$.
- If $f = \mathcal{O}(g)$ and $f = \Omega(g)$ then we use the notation $f = \Theta(g) \ \Rightarrow C_1.g(n) \leq f(n) \leq C_2.g(n)$ for $n \geq n_0, C_i \in \mathbb{R}^+$.

From Polynomial to Exponential Time

Definition

1. ***Polynomial time algorithm:*** *computational complexity is* $\mathcal{O}(n^k)$, *where* n *is the size of the input in bits and* $k \in \mathbb{R}^+$.

From Polynomial to Exponential Time

Definition

1. ***Polynomial time algorithm:*** *computational complexity is* $\mathcal{O}(n^k)$*, where* n *is the size of the input in bits and* $k \in \mathbb{R}^+$*.*
2. ***Exponential time algorithm:*** *computational complexity is of the form* $\mathcal{O}(c^{f(n)})$ *where* $c > 1$ *is a constant and* f *is a polynomial function on the size of the input* $n \in \mathbb{N}$*.*

From Polynomial to Exponential Time

Definition

1. ***Polynomial time algorithm:*** *computational complexity is* $\mathcal{O}(n^k)$, *where* n *is the size of the input in bits and* $k \in \mathbb{R}^+$.
2. ***Exponential time algorithm:*** *computational complexity is of the form* $\mathcal{O}(c^{f(n)})$ *where* $c > 1$ *is a constant and* f *is a polynomial function on the size of the input* $n \in \mathbb{N}$.
3. ***Subexponential time algorithm:*** *computational complexity for input* $q \in \mathbb{N}$[a] *is*

$$L_q(\alpha, c) = \mathcal{O}(e^{(c+o(1))(\ln q)^{\alpha}(\ln \ln q)^{1-\alpha}}),$$

where $\alpha \in \mathbb{R},\ 0 < \alpha < 1$ *and* c *is a positive constant.*

[a] Note that q is the input to the algorithm and not the size of the input.

Outline

1. Divisibility and Modular Arithmetic
2. Integer Representations and Algorithms
3. Primes and Greatest Common Divisors
4. Prime Numbers
5. Primes Generation

Primes

Definition

A positive integer $p > 1$ is called ***prime*** *if the only positive divisor of p are 1 and p.*

A positive integer $n > 1$ and is not prime is called ***composite****.*

Primes

Definition

*A positive integer $p > 1$ is called **prime** if the only positive divisor of p are 1 and p.*

*A positive integer $n > 1$ and is not prime is called **composite**.*

Lemma

Let p be a prime number, and suppose that $p \mid ab$. Then either $p \mid a$ or $p \mid b$ (or p divides both a and b).

Primes

Definition

A positive integer $p > 1$ is called ***prime*** *if the only positive divisor of p are 1 and p.*

A positive integer $n > 1$ and is not prime is called ***composite****.*

Lemma

Let p be a prime number, and suppose that $p \mid ab$. Then either $p \mid a$ or $p \mid b$ (or p divides both a and b).

Theorem

Let p be a prime number, and suppose that $p \mid a_1 a_2 \ldots a_r$. Then p divides at least one of the factors $a_1, a_2, \ldots, a_r$.

The Fundamental Theorem of Arithmetic

Theorem (The Fundamental Theorem of Arithmetic)

Every integer can be written as the product of primes (in order of nondecreasing size) in an essentially unique way.

The Fundamental Theorem of Arithmetic

Theorem (The Fundamental Theorem of Arithmetic)

Every integer can be written as the product of primes (in order of nondecreasing size) in an essentially unique way.

Every nonzero integer n can be expressed as a product of the form

$$n = \pm p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$$

where the p_i's are k distinct primes and the e_i's are integers with $e_i > 0$. This representation is ***unique*** *up to the order in which the factors are written*[a]*.*

[a] If we decide that 1 should be considered to be a prime, the uniqueness of this decomposition into primes would no longer hold!

The Fundamental Theorem of Arithmetic

The Fundamental Theorem of Arithmetic

Example

- $100 = 2.2.5.5 = 2^2.5^2$
- $641 = 641$
- $999 = 3.3.3.37 = 3^3.37$
- $1024 = 2.2.2.2.2.2.2.2.2.2 = 2^{10}$
- $9105293 =$

The Fundamental Theorem of Arithmetic

Example

- $100 = 2.2.5.5 = 2^2.5^2$
- $641 = 641$
- $999 = 3.3.3.37 = 3^3.37$
- $1024 = 2.2.2.2.2.2.2.2.2.2 = 2^{10}$
- $9105293 = 37 \times 43 \times 59 \times 97$

The Fundamental Theorem of Arithmetic

Example

- $100 = 2.2.5.5 = 2^2.5^2$
- $641 = 641$
- $999 = 3.3.3.37 = 3^3.37$
- $1024 = 2.2.2.2.2.2.2.2.2.2 = 2^{10}$
- $9105293 = 37 \times 43 \times 59 \times 97$

If n is not itself prime, then there must be a prime $p \leq \sqrt{n}$ that divides n.

The Fundamental Theorem of Arithmetic

Problem

1. *How can we tell if a given number n is prime or composite?*
2. *If n is composite, how can we factor it into primes?*

Revisit – Greatest Common Divisor (GCD)

Definition

Given $a, b \in \mathbb{Z},\ a \ \&\ b \neq 0$, *the* *greatest common divisor* $a \ \&\ b$, *denoted* $\gcd(a, b)$, *is the positive common divisor of* $a \ \&\ b$, *that is divisible by each of their common divisors. In other words, the largest integer* d *s/t* $d \mid a \ \&\ d \mid b$.

Definition

The integers a *and* b *are* *relatively prime* *if* $\gcd(a, b) = 1$.

Definition

The integers $a_1, a_2, \ldots, a_n$ *are* *pairwise relatively prime* *if* $\gcd(a_i, a_j) = 1$ *whenever* $1 \leq i < j \leq n$.

Revisit – GCD

- Suppose that the prime factorizations of the positive integers a & b are
$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n},$$
where each exponent is a nonnegative integer. Then
$$\gcd(a, b) = p_1^{min(a_1, b_1)} p_2^{min(a_2, b_2)} \dots p_n^{min(a_n, b_n)}$$

Revisit – GCD

- Suppose that the prime factorizations of the positive integers a & b are

$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n},$$

where each exponent is a nonnegative integer. Then

$$\gcd(a, b) = p_1^{min(a_1,b_1)} p_2^{min(a_2,b_2)} \dots p_n^{min(a_n,b_n)}$$

- Finding the $\gcd$ of two positive integers using their prime factorizations is not efficient because there is no efficient algorithm for finding the prime factorization of a positive integer.

Finding the Least Common Multiple (LCM)

Definition

The least common multiple of the positive integers a & b is the smallest positive integer that is divisible by both a & b. It is denoted by $lcm(a, b)$.

- Suppose

$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n},$$

where each exponent is a nonnegative integer. Then

$$lcm(a, b) = p_1^{max(a_1,b_1)} p_2^{max(a_2,b_2)} \dots p_n^{max(a_n,b_n)}$$

Finding the Least Common Multiple (LCM)

Definition

The least common multiple of the positive integers a & b is the smallest positive integer that is divisible by both a & b. It is denoted by $lcm(a, b)$.

- Suppose

$$a = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}, \quad b = p_1^{b_1} p_2^{b_2} \dots p_n^{b_n},$$

where each exponent is a nonnegative integer. Then

$$lcm(a, b) = p_1^{max(a_1, b_1)} p_2^{max(a_2, b_2)} \dots p_n^{max(a_n, b_n)}$$

Theorem

Let a & b be positive integers. Then

$$ab = \gcd(a, b) \times lcm(a, b)$$

Revisit – GCD

Theorem

(i) $\gcd(a, b) = \gcd(b, a)$.

(ii) $\gcd(a, a) = a$.

(iii) $\gcd(a, b) = \gcd(a - b, b)$

(iv) $\gcd(0, a) = a$.

Euclidean Algorithm

Euclidean algorithm for computing the $\gcd(a, b)$

Input: 2 non-negative integers a & b, with $a \geq b$.

Output: $\gcd(a, b)$

1. While $(b \neq 0)$ do
   1.1 Set $r \leftarrow a \; mod \; b$, $a \leftarrow b, \; b \leftarrow r$.
2. *Return*(a)

Euclidean Algorithm

Euclidean algorithm for computing the $\gcd(a, b)$

Input: 2 non-negative integers a & b, with $a \geq b$.

Output: $\gcd(a, b)$

1. While $(b \neq 0)$ do
   1.1 Set $r \leftarrow a \ mod \ b$, $a \leftarrow b, \ b \leftarrow r$.
2. *Return*(a)

$\gcd(4864, 3458)$

Euclidean Algorithm

Euclidean algorithm for computing the $\gcd(a, b)$

Input: 2 non-negative integers a & b, with $a \geq b$.
Output: $\gcd(a, b)$

1. While $(b \neq 0)$ do
   1.1 Set $r \leftarrow a \ mod \ b$, $a \leftarrow b, \ b \leftarrow r$.
2. *Return*(a)

$\gcd(4864, 3458)$

$$
\begin{aligned}
4864 &= 1.3458 + 1406 \\
3458 &= 2.1406 + 646 \\
1406 &= 2.646 + 114 \\
646 &= 5.114 + 76 \\
114 &= 1.76 + 38 \\
76 &= 2.38 + 0.
\end{aligned}
$$

Correctness of Euclidean Algorithm

Lemma

Let $a = bq + r$, *where* $a, b, q, \ \& \, r \in \mathbb{Z}$ *and* $r \geq 0$. *Then* $\gcd(a, b) = \gcd(b, r)$.

Correctness of Euclidean Algorithm

Lemma

Let $a = bq + r$, *where* $a, b, q, \ \& \ r \in \mathbb{Z}$ *and* $r \geq 0$. *Then* $\gcd(a, b) = \gcd(b, r)$.

Proof.

- Suppose that $d \mid a$ and $d \mid b$. Then d also divides $a - bq = r$. Hence, any common divisor of $a \ \& \ b$ must also be any common divisor of $b \ \& \ r$.
- Suppose that $d \mid b$ and $d \mid r$. Then $d \mid (bq + r) = a$. Hence, any common divisor of $a \ \& \ b$ must also be a common divisor of $b \ \& \ r$.
- Therefore, $\gcd(a, b) = \gcd(b, r)$.

□

GCDs as Linear Combinations

Bézout's Lemma

$\forall\ a, b \in \mathbb{Z},\ \exists\ s, t \in \mathbb{Z}$ s/t $gcd(a, b) = s.a + t.b$

Definition

If a & b are positive integers, then integers s & t s/t $\gcd(a, b) = sa + tb$ are called Bézout coefficients of a & b. The equation $\gcd(a, b) = sa + tb$ is called Bézout's identity.

GCDs as Linear Combinations

Bézout's Lemma

$\forall\ a, b \in \mathbb{Z},\ \exists\ s, t \in \mathbb{Z}$ s/t $gcd(a, b) = s.a + t.b$

Definition

If a & b are positive integers, then integers s & t s/t $\gcd(a, b) = sa + tb$ are called Bézout coefficients of a & b. The equation $\gcd(a, b) = sa + tb$ is called Bézout's identity.

- By Bézout's lemma, the $\gcd(a, b)$ can be expressed in the form $sa + tb$ where $s, t \in \mathbb{Z}$. This is a linear combination with integer coefficients of a & b.

GCDs as Linear Combinations

Bézout's Lemma

$\forall\ a, b \in \mathbb{Z},\ \exists\ s, t \in \mathbb{Z}$ s/t $gcd(a, b) = s.a + t.b$

Definition

If a & b are positive integers, then integers s & t s/t $\gcd(a, b) = sa + tb$ are called Bézout coefficients of a & b. The equation $\gcd(a, b) = sa + tb$ is called Bézout's identity.

- By Bézout's lemma, the $\gcd(a, b)$ can be expressed in the form $sa + tb$ where $s, t \in \mathbb{Z}$. This is a linear combination with integer coefficients of a & b.
- The smallest positive value of $sa + tb = \gcd(a, b)$

Extended Euclidean Algorithm

Extended Euclidean algorithm

Input: 2 non-negative integers a & b, with $a \geq b$.
Output: $d = \gcd(a, b)$ & $x, y \in \mathbb{Z}$ s/t $ax + by = d$.

1. If $b = 0$ then set $d \leftarrow a,\ x \leftarrow 1,\ y \leftarrow 0$, and $return(d, x, y)$.
2. Set $x_2 \leftarrow 1,\ x_1 \leftarrow 0,\ y_2 \leftarrow 0,\ y_1 \leftarrow 1$.
3. While $(b > 0)$ do
   - 3.1 $q \leftarrow \lfloor a/b \rfloor,\ r \leftarrow a - qb$, $x \leftarrow x_2 - qx_1,\ y \leftarrow y_2 - qy_1$.
   - 3.2 $a \leftarrow b,\ b \leftarrow r,\ x_2 \leftarrow x_1$, $x_1 \leftarrow x,\ y_2 \leftarrow y_1$, and $y_1 \leftarrow y$.
4. Set $d \leftarrow a,\ x \leftarrow x_2,\ y \leftarrow y_2$, and $return(d, x, y)$.

Extended Euclidean Algorithm

Extended Euclidean algorithm

Input: 2 non-negative integers a & b, with $a \geq b$.
Output: $d = \gcd(a, b)$ & $x, y \in \mathbb{Z}$ s/t $ax + by = d$.

1. If $b = 0$ then set $d \leftarrow a,\ x \leftarrow 1,\ y \leftarrow 0$, and $return(d, x, y)$.
2. Set $x_2 \leftarrow 1,\ x_1 \leftarrow 0,\ y_2 \leftarrow 0,\ y_1 \leftarrow 1$.
3. While $(b > 0)$ do
   - 3.1 $q \leftarrow \lfloor a/b \rfloor,\ r \leftarrow a - qb$, $x \leftarrow x_2 - qx_1,\ y \leftarrow y_2 - qy_1$.
   - 3.2 $a \leftarrow b,\ b \leftarrow r,\ x_2 \leftarrow x_1$, $x_1 \leftarrow x,\ y_2 \leftarrow y_1$, and $y_1 \leftarrow y$.
4. Set $d \leftarrow a,\ x \leftarrow x_2,\ y \leftarrow y_2$, and $return(d, x, y)$.

$a = 4864,\ b = 3458$

Extended Euclidean Algorithm

Extended Euclidean algorithm

Input: 2 non-negative integers a & b, with $a \geq b$.
Output: $d = \gcd(a, b)$ & $x, y \in \mathbb{Z}$ s/t $ax + by = d$.

1. If $b = 0$ then set $d \leftarrow a,\ x \leftarrow 1,\ y \leftarrow 0$, and $return(d, x, y)$.
2. Set $x_2 \leftarrow 1,\ x_1 \leftarrow 0,\ y_2 \leftarrow 0,\ y_1 \leftarrow 1$.
3. While $(b > 0)$ do
   - 3.1 $q \leftarrow \lfloor a/b \rfloor,\ r \leftarrow a - qb$, $x \leftarrow x_2 - qx_1,\ y \leftarrow y_2 - qy_1$.
   - 3.2 $a \leftarrow b,\ b \leftarrow r,\ x_2 \leftarrow x_1$, $x_1 \leftarrow x,\ y_2 \leftarrow y_1$, and $y_1 \leftarrow y$.
4. Set $d \leftarrow a,\ x \leftarrow x_2,\ y \leftarrow y_2$, and $return(d, x, y)$.

$a = 4864,\ b = 3458$

q	r	x	y	a	b	x_2	x_1	y_2	y_1
–	–	–	–	4864	3458	1	0	0	1
1	1406	1	−1	3458	1406	0	1	1	−1
2	646	−2	3	1406	646	1	−2	−1	3
2	114	5	−7	646	114	−2	5	3	−7
5	76	−27	38	114	76	5	−27	−7	38
1	38	32	−45	76	38	−27	32	38	−45
2	0	−91	128	38	0	32	−91	−45	128

$38 = 32.4864 - 45.3458$

Consequences of Bézout's Theorem

Lemma

If $a, b, c \in \mathbb{N}$ *s/t* $\gcd(a, b) = 1$ *and* $a \mid bc$*, then* $a \mid c$*.*

Lemma

If p *is prime and* $p \mid a_1 a_2 \ldots a_n$*, then* $p \mid a_i$ *for some* i*.*

Theorem

Let m *be a positive integer and let* $a, b, c \in \mathbb{Z}$*. If* $ac \equiv bc \mod m$ *and* $\gcd(c, m) = 1$*, then* $a \equiv b \mod m$*.*

Revisit – Congruences

- If $ac \equiv bc \mod m$, it need not be true that $a \equiv b \mod m$.
- It is not always possible to divide congruences.

Revisit – Congruences

- If $ac \equiv bc \mod m$, it need not be true that $a \equiv b \mod m$.
- It is not always possible to divide congruences.
- $15 \times 2 \equiv 20 \times 2 \mod 10$, however, $15 \not\equiv 20 \mod 10$.

Revisit – Congruences

- If $ac \equiv bc \mod m$, it need not be true that $a \equiv b \mod m$.
- It is not always possible to divide congruences.
- $15 \times 2 \equiv 20 \times 2 \mod 10$, however, $15 \not\equiv 20 \mod 10$.
- $uv \equiv 0 \mod m$ with $u \not\equiv 0 \mod m$ and $v \not\equiv 0 \mod m$.

Revisit – Congruences

- If $ac \equiv bc \mod m$, it need not be true that $a \equiv b \mod m$.
- It is not always possible to divide congruences.
- $15 \times 2 \equiv 20 \times 2 \mod 10$, however, $15 \not\equiv 20 \mod 10$.
- $uv \equiv 0 \mod m$ with $u \not\equiv 0 \mod m$ and $v \not\equiv 0 \mod m$.
- $6 \times 4 \equiv 0 \mod 12$, however, $6 \not\equiv 0 \mod 12$ and $4 \equiv 0 \mod 12$.

Revisit – Congruences

- If $ac \equiv bc \mod m$, it need not be true that $a \equiv b \mod m$.
- It is not always possible to divide congruences.
- $15 \times 2 \equiv 20 \times 2 \mod 10$, however, $15 \not\equiv 20 \mod 10$.
- $uv \equiv 0 \mod m$ with $u \not\equiv 0 \mod m$ and $v \not\equiv 0 \mod m$.
- $6 \times 4 \equiv 0 \mod 12$, however, $6 \not\equiv 0 \mod 12$ and $4 \equiv 0 \mod 12$.
- If $\gcd(c, m) = 1$, then we can cancel c from $ac \equiv bc \mod m$.

Revisit – Congruences

- Solve $x^2 + 2x - 1 \equiv 0 \mod 7$

Revisit – Congruences

- Solve $x^2 + 2x - 1 \equiv 0 \mod 7$

  $x \equiv 2 \mod 7$ and $x \equiv 3 \mod 7$ are the two solutions

- Solve $6x \equiv 15 \mod 514$.

Revisit – Congruences

- Solve $x^2 + 2x - 1 \equiv 0 \mod 7$

  $x \equiv 2 \mod 7$ and $x \equiv 3 \mod 7$ are the two solutions

- Solve $6x \equiv 15 \mod 514$.

  The congruence has no solutions.

Revisit – Congruences

- Solve $x^2 + 2x - 1 \equiv 0 \mod 7$

  $x \equiv 2 \mod 7$ and $x \equiv 3 \mod 7$ are the two solutions

- Solve $6x \equiv 15 \mod 514$.

  The congruence has no solutions.

Theorem

Let a, c, and m be integers with $m \geq 1$, and let $g = \gcd(a, m)$.

(i) *If $g \nmid c$, then the congruence $ax \equiv c \mod m$ has no solutions.*

(ii) *If $g \mid c$, then the congruence $ax \equiv c \mod m$ has exactly g incongruent solutions.*

Revisit – Linear Congruences

Definition

A congruence of the form

$$ax \equiv b \mod m,$$

where $m \in \mathbb{N}, a \;\&\; b \in \mathbb{Z}$, and x is a variable, is called a linear congruence.

Revisit – Linear Congruences

- One method of solving linear congruences is by finding the inverse $\bar{a} \mod m$, if it exists.
- Although we can not divide both sides of the congruence by a, we can multiply by $\bar{a}$ to solve for x.

Revisit – Linear Congruences

- One method of solving linear congruences is by finding the inverse $\bar{a} \mod m$, if it exists.
- Although we can not divide both sides of the congruence by a, we can multiply by $\bar{a}$ to solve for x.

Theorem

If a & m are relatively prime integers and $m > 1$, then an inverse of a modulo m exists. Furthermore, this inverse is unique modulo m.

Revisit – Linear Congruences

Theorem

Let $a, m \in \mathbb{Z}$ *with* $m > 0$, *and let* $d := \gcd(a, m)$.

1. *For every* $b \in \mathbb{Z}$, *the congruence* $ax \equiv b \mod m$ *has a solution iff* $d \mid b$.
2. *For every* $x \in \mathbb{Z}$, *we have* $ax \equiv 0 \mod m$ *iff* $x \equiv 0 \mod \frac{m}{d}$.
3. *For all* $x, x' \in \mathbb{Z}$, *we have* $ax \equiv ax' \mod m$ *iff* $x \equiv x' \mod \frac{m}{d}$

Revisit – Linear Congruences

Example

In the following table is an illustration for $m = 15$ *and* $a = 1, 2, 3, 4, 5$.

$1.x$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$2.x$	0	2	4	6	8	10	12	14	1	3	5	7	9	11	13
$3.x$	0	3	6	9	12	0	3	6	9	12	0	3	6	9	12
$4.x$	0	4	8	12	1	5	9	13	2	6	10	14	3	7	11
$5.x$	0	5	10	0	5	10	0	5	10	0	5	10	0	5	10

Revisit – Congruences

Theorem

Let p be a prime number and let

$$f(x) = a_0 x^d + a_1 x^{d-1} + \cdots + a_d$$

be a polynomial of degree $d \geq 1$ with integer coefficients and with $p \nmid a_0$.
Then the congruence

$$f(x) \equiv 0 \mod p$$

has at most d incongruent solutions.

Fermat's Little Theorem

Fermat's Little Theorem

- Take a non-zero number $a \in \mathbb{Z}_m$ and compute its powers $a, a^2, a^3, \ldots a^m \mod m$.

Fermat's Little Theorem

- Take a non-zero number $a \in \mathbb{Z}_m$ and compute its powers $a, a^2, a^3, \ldots a^m \mod m$.

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

Fermat's Little Theorem

- Take a non-zero number $a \in \mathbb{Z}_m$ and compute its powers $a, a^2, a^3, \ldots a^m \mod m$.

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

- Use Fermat's Little Theorem to simplify computations

$$6^{22} - 1 =$$

Fermat's Little Theorem

- Take a non-zero number $a \in \mathbb{Z}_m$ and compute its powers $a, a^2, a^3, \ldots a^m \mod m$.

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

- Use Fermat's Little Theorem to simplify computations

$$6^{22} - 1 = 23 \times 5722682775750745.$$

$$2^{35} \mod 7$$

Fermat's Little Theorem

- Take a non-zero number $a \in \mathbb{Z}_m$ and compute its powers $a, a^2, a^3, \ldots a^m \mod m$.

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

- Use Fermat's Little Theorem to simplify computations

$$6^{22} - 1 = 23 \times 5722682775750745.$$

$$2^{35} \mod 7 \equiv 32 \equiv 4 \mod 7.$$

Fermat's Little Theorem

Lemma

Let p be a prime number and let a be a number s/t $a \not\equiv 0 \mod p$. Then the numbers

$$a, 2a, 3a, \ldots, (p-1)a \mod p$$

are the same as the numbers

$$1, 2, 3, \ldots, (p-1) \mod p,$$

although they may be in a different order.

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

- Fermat's Little Theorem can be used to show that a number is not a prime without actually factoring it.

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

- Fermat's Little Theorem can be used to show that a number is not a prime without actually factoring it.
- E.g.,

$$2^{1234566} \equiv 899557 \mod 1234567.$$

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

- Fermat's Little Theorem can be used to show that a number is not a prime without actually factoring it.
- E.g.,

$$2^{1234566} \equiv 899557 \mod 1234567.$$

This means that $1234567 (= 127 \times 9721)$ cannot be a prime.

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

- Fermat's Little Theorem can be used to show that a number is not a prime without actually factoring it.
- E.g.,

$$2^{1234566} \equiv 899557 \mod 1234567.$$

This means that $1234567(= 127 \times 9721)$ cannot be a prime.

- Consider the number $m = 10^{100} + 37$.

Fermat's Little Theorem

Theorem

Let p be a prime number, and let a be any number s/t $a \not\equiv 0 \mod p$. Then

$$a^{p-1} \equiv 1 \mod p.$$

- Fermat's Little Theorem can be used to show that a number is not a prime without actually factoring it.
- E.g.,

$$2^{1234566} \equiv 899557 \mod 1234567.$$

This means that $1234567 (= 127 \times 9721)$ cannot be a prime.

- Consider the number $m = 10^{100} + 37$. Verify $2^{m-1} \not\equiv 1 \mod m$.

Euler's Generalization

- Fermat's Little Theorem is certainly not true if we replace p by a composite number.

Euler's Generalization

- Fermat's Little Theorem is certainly not true if we replace p by a composite number.

$$5^5 \mod 6 \equiv$$

Euler's Generalization

- Fermat's Little Theorem is certainly not true if we replace p by a composite number.

$$5^5 \mod 6 \equiv 5 \mod 6, \quad 2^8 \mod 9 \equiv$$

Euler's Generalization

- Fermat's Little Theorem is certainly not true if we replace p by a composite number.

$$5^5 \mod 6 \equiv 5 \mod 6, \quad 2^8 \mod 9 \equiv 4 \mod 9.$$

- Can we find x s/t

$$a^x \equiv 1 \mod m.$$

Euler's Generalization

- Fermat's Little Theorem is certainly not true if we replace p by a composite number.

$$5^5 \mod 6 \equiv 5 \mod 6, \quad 2^8 \mod 9 \equiv 4 \mod 9.$$

- Can we find x s/t

$$a^x \equiv 1 \mod m.$$

- **Claim:** $\nexists x$ if $\gcd(a, m) > 1$.

Euler's Generalization

- The number of integers between 1 and m that are relatively prime to m is denoted by $\phi(m)$ and is defined by

$$\phi(m) = \#\{a : 1 \leq a \leq m \text{ and } \gcd(a,m) = 1\}.$$

$$\phi(m) = \sum^{m}_{\substack{k=1 \\ \gcd(k,m)=1}} 1$$

The function $\phi(\cdot)$ is called Euler's phi function.

Euler's Generalization

Lemma

Let

$$1 \leq b_1 < b_2 < \cdots < b_{\phi(m)} < m.$$

be the $\phi(m)$ numbers between 0 and m that are relatively prime to m. If $\gcd(a, m) = 1$, then the numbers

$$b_1 a, b_2 a, b_3 a, \ldots, b_{\phi(m)} a \mod m$$

are the same as the numbers

$$b_1, b_2, b_3, \ldots, b_{\phi(m)} \mod m,$$

although they may be in a different order.

Euler's Theorem

Theorem

If $\gcd(a, m) = 1$, *then*

$$a^{\phi(m)} \equiv 1 \mod m.$$

Euler's Theorem

Theorem

If $\gcd(a, m) = 1$, *then*

$$a^{\phi(m)} \equiv 1 \mod m.$$

- It is a beautiful and powerful result,

Euler's Theorem

Theorem

If $\gcd(a, m) = 1$, *then*

$$a^{\phi(m)} \equiv 1 \mod m.$$

- It is a beautiful and powerful result, however, it will not be of much use if computing $\phi(m)$ is hard.
- Compute $\phi(1000) =$

Euler's Theorem

Theorem

If $\gcd(a, m) = 1$, *then*

$$a^{\phi(m)} \equiv 1 \mod m.$$

- It is a beautiful and powerful result, however, it will not be of much use if computing $\phi(m)$ is hard.
- Compute $\phi(1000) = 400$
- Compute $\phi(10^{100}) =$

Euler's Theorem

Theorem

If $\gcd(a, m) = 1$, *then*

$$a^{\phi(m)} \equiv 1 \mod m.$$

- It is a beautiful and powerful result, however, it will not be of much use if computing $\phi(m)$ is hard.
- Compute $\phi(1000) = 400$
- Compute $\phi(10^{100}) = 4 \times 10^{99}$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) = \phi(7^4) = (7^4 - 7^3) = 2058$

(ii) $\phi(14) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) = \phi(7^4) = (7^4 - 7^3) = 2058$

(ii) $\phi(14) = 6$

(iii) $\phi(15) =$

Euler's phi Function

Properties of Euler's phi function

1. If p is a prime, then $\phi(p) = p - 1$.
2. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) = \phi(7^4) = (7^4 - 7^3) = 2058$

(ii) $\phi(14) = 6$

(iii) $\phi(15) = 8$

(iv) $\phi(210) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) = \phi(7^4) = (7^4 - 7^3) = 2058$

(ii) $\phi(14) = 6$

(iii) $\phi(15) = 8$

(iv) $\phi(210) = \phi(14 \times 15) =$

Euler's phi Function

Properties of Euler's phi function

i. If p is a prime, then $\phi(p) = p - 1$.

ii. If p is a prime, then $\phi(p^m) = (p^m - p^{m-1})$.

Example

Compute

(i) $\phi(2401) = \phi(7^4) = (7^4 - 7^3) = 2058$

(ii) $\phi(14) = 6$

(iii) $\phi(15) = 8$

(iv) $\phi(210) = \phi(14 \times 15) = 48$

Euler's phi Function

Properties of Euler's phi function

iii. The Euler phi function is **multiplicative**. That is, if $gcd(m, n) = 1$, then $\phi(mn) = \phi(m)\phi(n)$.

Euler's phi Function

Properties of Euler's phi function

iii. The Euler phi function is **multiplicative**. That is, if $gcd(m, n) = 1$, then $\phi(mn) = \phi(m)\phi(n)$.

- Let $S = \{a : 1 \leq a \leq mn \text{ and } \gcd(a, mn) = 1\}$.
- Let

$$T = \left\{ (b, c) \quad : \quad \begin{array}{c} 1 \leq b \leq m \text{ and } \gcd(b, m) = 1 \\ \\ 1 \leq c \leq n \text{ and } \gcd(c, n) = 1 \end{array} \right\}$$

Euler's phi Function

Properties of Euler's phi function

iii. The Euler phi function is **multiplicative**. That is, if $gcd(m, n) = 1$, then $\phi(mn) = \phi(m)\phi(n)$.

- Let $S = \{a : 1 \leq a \leq mn \text{ and } \gcd(a, mn) = 1\}$.
- Let

$$T = \left\{ (b, c) \;:\; \begin{array}{c} 1 \leq b \leq m \text{ and } \gcd(b, m) = 1 \\ \\ 1 \leq c \leq n \text{ and } \gcd(c, n) = 1 \end{array} \right\}$$

$$a \mod mn \mapsto (a \mod m, a \mod n)$$

Euler's phi Function

1. To prove different numbers in S map to to different pairs in T.
2. Every pair in T maps to some number in S.

Euler's phi Function

1. To prove different numbers in S map to to different pairs in T.
2. Every pair in T maps to some number in S.

Theorem (Chinese Remainder Theorem (CRT))

Let m and n be integers satisfying $\gcd(m,n)=1$, and let b and c be any integers. Then the simultaneous congruences

$$x \equiv b \mod m \quad \text{and} \quad x \equiv c \mod n$$

have $!$ solution in $0 \leq x < mn$.

Chinese Remainder Theorem

Example

Solve

$$x \equiv 8 \mod 11 \quad \text{and} \quad x \equiv 3 \mod 19.$$

Chinese Remainder Theorem

Example

Solve

$$x \equiv 8 \mod 11 \quad \text{and} \quad x \equiv 3 \mod 19.$$

Chinese Remainder Theorem

- In the first century, the Chinese mathematician Sun-Tzu asked: *There are certain things whose number is unknown. When divided by 3, the remainder is 2; when divided by 5, the remainder is 3; when divided by 7, the remainder is 2. What will be the number of things?*

Chinese Remainder Theorem

- In the first century, the Chinese mathematician Sun-Tzu asked: *There are certain things whose number is unknown. When divided by 3, the remainder is 2; when divided by 5, the remainder is 3; when divided by 7, the remainder is 2. What will be the number of things?*
- This puzzle can be translated into the solution of the system of congruences:

$$x \equiv 2 \mod 3,$$
$$x \equiv 3 \mod 5,$$
$$x \equiv 2 \mod 7?$$

- Now, we'll see how the Chinese Remainder Theorem can be used to solve Sun-Tzu's problem.

Chinese Remainder Theorem

Theorem (CRT)

If the integers $n_1, n_2, \cdots, n_k$ *are pairwise relatively prime, then the system of simultaneous congruences*

$$x \equiv a_i \ mod \ n_i,$$

for $1 \leq i \leq k$ *has a* $!$ *solution modulo* $n = n_1 n_2 \cdots n_k$ *which is given by*

Chinese Remainder Theorem

Theorem (CRT)

If the integers $n_1, n_2, \cdots, n_k$ *are pairwise relatively prime, then the system of simultaneous congruences*

$$x \equiv a_i \; mod \; n_i,$$

for $1 \leq i \leq k$ *has a* $!$ *solution modulo* $n = n_1 n_2 \cdots n_k$ *which is given by*

$$x = \sum_{i=1}^{k} a_i N_i M_i \; mod \; n,$$

where $N_i = n/n_i \;\&\; M_i = N_i^{-1} \; mod \; n_i$.

Chinese Remainder Theorem

Example

Consider the 3 congruences from Sun-Tzu's problem:
$x \equiv 2 \mod 3, \quad x \equiv 3 \mod 5, \quad x \equiv 2 \mod 7$.

- $n = 3.5.7 = 105, \; N_1 = n/3 = 35, \; N_2 = 21, \; \& \; N_3 = 15$

Chinese Remainder Theorem

Example

Consider the 3 congruences from Sun-Tzu's problem:
$x \equiv 2 \mod 3, \quad x \equiv 3 \mod 5, \quad x \equiv 2 \mod 7$.

- $n = 3.5.7 = 105,\ N_1 = n/3 = 35,\ N_2 = 21,\ \&\ N_3 = 15$

Euler's phi Function

Properties of Euler's phi function

iv. If $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$, is the prime factorization of n, then

$$\phi(n) \quad =$$

Euler's phi Function

Properties of Euler's phi function

iv. If $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$, is the prime factorization of n, then

$$\begin{aligned}\phi(n) &= \left(p_1^{e_1} - p_1^{e_1-1}\right)\left(p_2^{e_2} - p_2^{e_2-1}\right)\ldots\left(p_k^{e_k} - p_k^{e_k-1}\right) \\ &= n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\ldots\left(1 - \frac{1}{p_k}\right).\end{aligned}$$

Outline

1 Divisibility and Modular Arithmetic

2 Integer Representations and Algorithms

3 Primes and Greatest Common Divisors

4 Prime Numbers

5 Primes Generation

Infinitude of Primes

Theorem (Euclid)

There are infinitely many primes.

Infinitude of Primes

Theorem (Euclid)

There are infinitely many primes.

Proof.

- Assume there are finitely many primes: $p_1, p_2, \ldots, p_n$
- Let $q = p_1 p_2 \ldots p_n + 1$

Infinitude of Primes

Theorem (Euclid)

There are infinitely many primes.

Proof.

- Assume there are finitely many primes: $p_1, p_2, \ldots, p_n$
- Let $q = p_1 p_2 \ldots p_n + 1$
- Either q is prime or by the fundamental theorem of arithmetic it is a product of primes.
- However $p_j \nmid q$ for $1 \leq j \leq n$;

Infinitude of Primes

Theorem (Euclid)

There are infinitely many primes.

Proof.

- Assume there are finitely many primes: $p_1, p_2, \ldots, p_n$
- Let $q = p_1 p_2 \ldots p_n + 1$
- Either q is prime or by the fundamental theorem of arithmetic it is a product of primes.
- However $p_j \nmid q$ for $1 \leq j \leq n$; since if $p_j \mid q$, then $p_j \mid (q - p_1 p_2 \ldots p_n) \quad \Rightarrow \quad p_j \mid 1$
- Hence, there is a prime q not on the list $p_1, p_2, \ldots, p_n$.

Infinitude of Primes

Theorem (Euclid)

There are infinitely many primes.

Proof.

- Assume there are finitely many primes: $p_1, p_2, \ldots, p_n$
- Let $q = p_1 p_2 \ldots p_n + 1$
- Either q is prime or by the fundamental theorem of arithmetic it is a product of primes.
- However $p_j \nmid q$ for $1 \le j \le n$; since if $p_j \mid q$, then $p_j \mid (q - p_1 p_2 \ldots p_n) \quad \Rightarrow \quad p_j \mid 1$
- Hence, there is a prime q not on the list $p_1, p_2, \ldots, p_n$.

□

Note: This proof was given by Euclid in The Elements more than 2000 years ago. The proof is considered to be one of the most beautiful in all mathematics. It is the first proof in The Book, inspired by the famous mathematician Paul Erdös imagined collection of perfect proofs maintained by God.

Infinitude of Primes

Example

We start with a list consisting of the single prime $\{2\}$[a]. Then we compute

$$
\begin{aligned}
n &= 2 + 1 = 3 && \rightarrow \text{prime} \\
n &= 2.3 + 1 = 7 && \rightarrow \text{prime} \\
n &= 2.3.7 + 1 = 43 && \rightarrow \text{prime} \\
n &= 2.3.7.43 + 1 = 1807
\end{aligned}
$$

Infinitude of Primes

Example

We start with a list consisting of the single prime $\{2\}$[a]. Then we compute

$$\begin{aligned} n &= 2+1=3 && \rightarrow \text{prime} \\ n &= 2.3+1=7 && \rightarrow \text{prime} \\ n &= 2.3.7+1=43 && \rightarrow \text{prime} \\ n &= 2.3.7.43+1=1807=13\times 139 && \rightarrow \text{not prime} \end{aligned}$$

[a] 2 is the oddest prime!

Infinitude of Primes

- Every odd number is congruent to either 1 or $3 \mod 4$

Infinitude of Primes

- Every odd number is congruent to either 1 or 3 $\mod 4$

Odd Primes

$1 \; mod \; 4 \quad 3 \; mod \; 4$

Infinitude of Primes

Theorem

There are infinitely many primes of the form $3 \mod 4$.

Infinitude of Primes

Theorem

There are infinitely many primes of the form $3 \mod 4$.

Infinitude of Primes

Theorem

There are infinitely many primes of the form $1 \mod 4$.

Infinitude of Primes

Theorem

There are infinitely many primes of the form $1 \mod 4$.

Infinitude of Primes

Theorem (Dirichlet's Theorem on Primes in Arithmetic Progressions)

Let a and m be integers with $\gcd(a, m) = 1$. Then there are infinitely many primes of the form

$$p \equiv a \mod m.$$

The Prime Number Theorem

The Prime Number Theorem

Theorem

When x is large, the number of primes less than $x \approx \frac{x}{ln(x)}$. In other words,

$$\lim_{x \to \infty} \frac{\pi(x)}{x/ln(x)} = 1,$$

where

$$\pi(x) = \#\{primes\ p\ with\ p \leq x\}$$

Conjectures

Conjecture (Goldbach's Conjecture)

Every even number $n \geq 4$ is a sum of two primes.

Conjectures

Conjecture (Goldbach's Conjecture)

Every even number $n \geq 4$ is a sum of two primes.

Conjecture (The Twin Primes Conjecture)

There are infinitely many prime numbers p s/t $p + 2$ is also prime.

Conjectures

Conjecture (Goldbach's Conjecture)

Every even number $n \geq 4$ is a sum of two primes.

Conjecture (The Twin Primes Conjecture)

There are infinitely many prime numbers p s/t $p + 2$ is also prime.

Conjecture (The $n^2 + 1$ Conjecture)

There are infinitely many primes of the form $n^2 + 1$

Mersenne Primes

- Let $m = a^n - 1$, for $n \geq 2$. $m \in \{prime,\ composite\}$.

Mersenne Primes

- Let $m = a^n - 1$, for $n \geq 2$. $m \in \{prime,\ composite\}$.
- $$x^n - 1 = (x-1)(x^{n-1} + x^{n-2} + \cdots + x^2 + x + 1).$$
- $(a-1) \mid (a^n - 1)$. So $a^n - 1$ will be composite unless $a - 1 = 1 \Rightarrow a = 2$.
- **Observation:**
  - (i) $2^n - 1$ is divisible by 3, when n is even.

Mersenne Primes

- Let $m = a^n - 1$, for $n \geq 2$. $m \in \{prime,\ composite\}$.
- $$x^n - 1 = (x-1)(x^{n-1} + x^{n-2} + \cdots + x^2 + x + 1).$$
- $(a-1) \mid (a^n - 1)$. So $a^n - 1$ will be composite unless $a - 1 = 1 \Rightarrow a = 2$.
- **Observation:**
  1. (i) $2^n - 1$ is divisible by 3, when n is even.
  2. (ii) $2^n - 1$ is divisible by 7, when n is divisible by 3
  3. (iii) $2^n - 1$ is divisible by 31, when n is divisible by 5

Mersenne Primes

Proposition

If $a^n - 1$ is prime for some numbers $a \geq 2$ and $n \geq 2$, then a must equal 2 and n must be a prime.

Mersenne Primes

Proposition

If $a^n - 1$ is prime for some numbers $a \geq 2$ and $n \geq 2$, then a must equal 2 and n must be a prime.

- If we are interested in primes of the form $a^n - 1$ we only need to a number of the form

  $2^p - 1$, where p is prime.

Mersenne Primes

Proposition

If $a^n - 1$ is prime for some numbers $a \geq 2$ and $n \geq 2$, then a must equal 2 and n must be a prime.

- If we are interested in primes of the form $a^n - 1$ we only need to a number of the form

$$2^p - 1, \text{ where } p \text{ is prime.}$$

Definition (Mersenne Primes)

Primes of the form $2^p - 1$ are called Mersenne primes.

Mersenne Primes

Proposition

If $a^n - 1$ is prime for some numbers $a \geq 2$ and $n \geq 2$, then a must equal 2 and n must be a prime.

- If we are interested in primes of the form $a^n - 1$ we only need to a number of the form

  $2^p - 1$, where p is prime.

Definition (Mersenne Primes)

Primes of the form $2^p - 1$ are called Mersenne primes.

The most recent Mersenne primes found in Dec 2018

$$M_{51} = 2^{82589933} - 1$$

Mersenne Primes

Proposition

If $a^n - 1$ is prime for some numbers $a \geq 2$ and $n \geq 2$, then a must equal 2 and n must be a prime.

- If we are interested in primes of the form $a^n - 1$ we only need to a number of the form

  $2^p - 1$, where p is prime.

Definition (Mersenne Primes)

Primes of the form $2^p - 1$ are called Mersenne primes.

The most recent Mersenne primes found in Dec 2018

$M_{51} = 2^{82589933} - 1 \to 24862048$-digit

Mersenne Primes

Open Problem

Are there infinitely many Mersenne primes, or does the list of Mersenne primes eventually stop?

Mersenne Primes

Open Problem

Are there infinitely many Mersenne primes, or does the list of Mersenne primes eventually stop?

Theorem (Euclid's Perfect Number Formula)

If $2^p - 1$ is a prime number, then $2^{p-1}(2^p - 1)$ is a perfect number.

Mersenne Primes

Open Problem

Are there infinitely many Mersenne primes, or does the list of Mersenne primes eventually stop?

Theorem (Euclid's Perfect Number Formula)

If $2^p - 1$ is a prime number, then $2^{p-1}(2^p - 1)$ is a perfect number.

Example

p	2	3	5	7	13
$2^{p-1}(2^p - 1)$	6	28			

Mersenne Primes

Open Problem

Are there infinitely many Mersenne primes, or does the list of Mersenne primes eventually stop?

Theorem (Euclid's Perfect Number Formula)

If $2^p - 1$ is a prime number, then $2^{p-1}(2^p - 1)$ is a perfect number.

Example

p	2	3	5	7	13
$2^{p-1}(2^p - 1)$	6	28	496	8128	33550336

σ Function

σ Function

Definition

This function $\sigma(n)$ is defined as

$$\sigma(n) = \text{ sum of all divisors of } n \text{ (including } 1 \text{ and } n).$$

σ Function

Definition

This function $\sigma(n)$ is defined as

$$\sigma(n) = \text{ sum of all divisors of } n \text{ (including } 1 \text{ and } n).$$

Example

$$\sigma(6)$$

σ Function

Definition

This function $\sigma(n)$ is defined as

$$\sigma(n) = \text{ sum of all divisors of } n \text{ (including } 1 \text{ and } n).$$

Example

$$\begin{aligned}
\sigma(6) &= 1+2+3+6 &= 12 \\
\sigma(8) &= 1+2+4+8 &= 15 \\
\sigma(18) &=
\end{aligned}$$

σ Function

Definition

This function $\sigma(n)$ is defined as

$$\sigma(n) = \text{ sum of all divisors of } n \text{ (including } 1 \text{ and } n).$$

Example

$$\begin{aligned}\sigma(6) &= 1+2+3+6 &&= 12\\ \sigma(8) &= 1+2+4+8 &&= 15\\ \sigma(18) &= 1+2+3+6+9+18 &&= 39\end{aligned}$$

Properties of σ Function

(i) $\sigma(p) =$

Properties of σ Function

(i) $\sigma(p) = p + 1$

(ii)

$$\sigma(p^k) = 1 + p + p^2 + \cdots + p^k =$$

Properties of σ Function

1. $\sigma(p) = p + 1$
2. $$\sigma(p^k) = 1 + p + p^2 + \cdots + p^k = \frac{p^{k+1} - 1}{p - 1}.$$
3. If $\gcd(m, n) = 1$, then $\sigma(mn) =$

Properties of σ Function

(i) $\sigma(p) = p + 1$

(ii)

$$\sigma(p^k) = 1 + p + p^2 + \cdots + p^k = \frac{p^{k+1} - 1}{p - 1}.$$

(iii) If $\gcd(m, n) = 1$, then $\sigma(mn) = \sigma(m)\sigma(n)$.

Example

- $\sigma(21) = 1 + 3 + 7 + 21 = (1 + 3) + 7(1 + 3) = (1 + 3)(1 + 7) = \sigma(3)\sigma(7)$
- $\sigma(30)$

Properties of σ Function

(i) $\sigma(p) = p + 1$

(ii)

$$\sigma(p^k) = 1 + p + p^2 + \cdots + p^k = \frac{p^{k+1} - 1}{p - 1}.$$

(iii) If $\gcd(m, n) = 1$, then $\sigma(mn) = \sigma(m)\sigma(n)$.

Example

- $\sigma(21) = 1 + 3 + 7 + 21 = (1 + 3) + 7(1 + 3) = (1 + 3)(1 + 7) = \sigma(3)\sigma(7)$
- $\sigma(30) = 1 + 2 + 3 + 5 + 6 + 10 + 15 + 30 = 72$
- $\sigma(5) = (5 + 1) = 6, \qquad \sigma(6) = 12$

Perfect Number

- How is the σ function related to perfect numbers?

Perfect Number

- How is the σ function related to perfect numbers?
- $\sigma(n) = 2n$, when n is perfect

Perfect Number

- How is the σ function related to perfect numbers?
- $\sigma(n) = 2n$, when n is perfect

Theorem (Euler's Perfect Number Theorem)

If n is an even perfect number, then n looks like

$$2^{p-1}(2^p - 1),$$

where $2^p - 1$ is a Mersenne prime.

Perfect Number

Perfect Number

Are there any odd perfect numbers?

Perfect Number

Are there any odd perfect numbers?

- There are no odd perfect numbers $< 10^{300}$.

Perfect Number

Are there any odd perfect numbers?

- There are no odd perfect numbers $< 10^{300}$. (till date)
- $\sigma(15) = \sigma(3) \times \sigma(5) = 24 < 2 \times 15$
- $\sigma(n) < 2n$ for odd n.

Perfect Number

Are there any odd perfect numbers?

- There are no odd perfect numbers $< 10^{300}$. (till date)
- $\sigma(15) = \sigma(3) \times \sigma(5) = 24 < 2 \times 15$
- $\sigma(n) < 2n$ for odd n.
- $n = 945 = 3^3 \times 5 \times 7 \Rightarrow \sigma(n) =$

Perfect Number

Are there any odd perfect numbers?

- There are no odd perfect numbers $< 10^{300}$. (till date)
- $\sigma(15) = \sigma(3) \times \sigma(5) = 24 < 2 \times 15$
- $\sigma(n) < 2n$ for odd n.
- $n = 945 = 3^3 \times 5 \times 7 \Rightarrow \sigma(n) = 1920 > 2n$

Powers $\mod m$

- We know how to compute

$$a^k \mod m,$$

efficiently.

Powers $\mod m$

- We know how to compute

$$a^k \mod m,$$

efficiently.

- Compute $5^{100000000000000} \mod 12830603$

Powers $\mod m$

- We know how to compute

$$a^k \mod m,$$

efficiently.

- Compute $5^{100000000000000} \mod 12830603$

$$12830603 = 3571 \times 3593 \Rightarrow \phi(12830603) = 12823440.$$

Powers $\mod m$

- We know how to compute
$$a^k \mod m,$$
efficiently.
- Compute $5^{100000000000000} \mod 12830603$

$$12830603 = 3571 \times 3593 \Rightarrow \phi(12830603) = 12823440.$$

$$100000000000000 = 7798219 \times 12823440 + 6546640$$

k^{th} Roots $\mod m$

- Now, how to find x efficiently when

$$x^k \equiv b \mod m$$

k^{th} Roots $\mod m$

- Now, how to find x efficiently when

$$x^k \equiv b \mod m \Rightarrow x \equiv \sqrt[k]{b} \mod m$$

k^{th} Roots $\mod m$

- Now, how to find x efficiently when

$$x^k \equiv b \mod m \Rightarrow x \equiv \sqrt[k]{b} \mod m$$

- Compute

$$\sqrt[4]{7} \mod 15$$

k^{th} Roots $\mod m$

- Now, how to find x efficiently when

$$x^k \equiv b \mod m \Rightarrow x \equiv \sqrt[k]{b} \mod m$$

- Compute

$$\sqrt[4]{7} \mod 15$$

- Compute

$$\sqrt[7]{22} \mod 33$$

k^{th} Roots $\mod m$

k^{th} roots $\mod m$

Let b, k, and m be given integers s/t $gcd(b, m) = 1$ and $\gcd(k, \phi(m)) = 1$
We can find a solution to the congruence

$$x^k \equiv b \mod m.$$

k^{th} Roots $\mod m$

k^{th} roots $\mod m$

Let b, k, and m be given integers s/t $gcd(b, m) = 1$ and $\gcd(k, \phi(m)) = 1$
We can find a solution to the congruence

$$x^k \equiv b \mod m.$$

(i) Compute $\phi(m)$.

(ii) Find positive integers u and v that satisfy $ku - \phi(m)v = 1$.

k^{th} Roots $\mod m$

k^{th} roots $\mod m$

Let b, k, and m be given integers s/t $gcd(b, m) = 1$ and $\gcd(k, \phi(m)) = 1$
We can find a solution to the congruence

$$x^k \equiv b \mod m.$$

(i) Compute $\phi(m)$.

(ii) Find positive integers u and v that satisfy $ku - \phi(m)v = 1$.

(iii) Compute $b^u \mod m$. The value obtained gives the solution x.

k^{th} Roots $\mod m$

Exercise

1. *Compute*

$$\sqrt[7]{2} \mod 33$$

k^{th} Roots $\mod m$

Exercise

1. *Compute*

$$\sqrt[7]{2} \mod 33 \Rightarrow 8 \equiv \sqrt[7]{2} \mod 33$$

2. *Compute*

$$\sqrt[11]{7} \mod 40$$

k^{th} Roots $\mod m$

Exercise

1. *Compute*

$$\sqrt[7]{2} \mod 33 \Rightarrow 8 \equiv \sqrt[7]{2} \mod 33$$

2. *Compute*

$$\sqrt[11]{7} \mod 40 \Rightarrow 23 \equiv \sqrt[11]{7} \mod 40$$

Outline

1. Divisibility and Modular Arithmetic
2. Integer Representations and Algorithms
3. Primes and Greatest Common Divisors
4. Prime Numbers
5. Primes Generation

The Sieve of Erastosthenes

The Sieve of Erastosthenes

- The Sieve of Erastosthenes can be used to find all primes not exceeding a specified positive integer n.

  For example, begin with the list of integers between 1 and 100.

  (i) Delete all the integers, other than 2, divisible by 2.
  (ii) Delete all the integers, other than 3, divisible by 3.
  (iii) Next, delete all the integers, other than 5, divisible by 5.
  (iv) Next, delete all the integers, other than 7, divisible by 7.
  (v) Since all the remaining integers are not divisible by any of the previous integers, other than 1, the primes are:

  $$\{2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,$$

  $$71,73,79,83,89,97\}$$

The Sieve of Erastosthenes

- The Sieve of Erastosthenes can be used to find all primes not exceeding a specified positive integer n.

  For example, begin with the list of integers between 1 and 100.

  (i) Delete all the integers, other than 2, divisible by 2.
  (ii) Delete all the integers, other than 3, divisible by 3.
  (iii) Next, delete all the integers, other than 5, divisible by 5.
  (iv) Next, delete all the integers, other than 7, divisible by 7.
  (v) Since all the remaining integers are not divisible by any of the previous integers, other than 1, the primes are:

  $$\{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67,$$
  $$71, 73, 79, 83, 89, 97\}$$

- Computational complexity of this algo

The Sieve of Erastosthenes

- The Sieve of Erastosthenes can be used to find all primes not exceeding a specified positive integer n.

  For example, begin with the list of integers between 1 and 100.

  (i) Delete all the integers, other than 2, divisible by 2.
  (ii) Delete all the integers, other than 3, divisible by 3.
  (iii) Next, delete all the integers, other than 5, divisible by 5.
  (iv) Next, delete all the integers, other than 7, divisible by 7.
  (v) Since all the remaining integers are not divisible by any of the previous integers, other than 1, the primes are:

  $$\{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67,$$
  $$71, 73, 79, 83, 89, 97\}$$

- Computational complexity of this algo $= O(n \log \log n)$

The Sieve of Erastosthenes

All prime numbers in the range [1 : 16]

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	3	~~4~~	5	~~6~~	7	~~8~~	9	~~10~~	11	~~12~~	13	~~14~~	15	~~16~~
2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~	11	~~12~~	13	~~14~~	~~15~~	~~16~~
2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~	11	~~12~~	13	~~14~~	~~15~~	~~16~~
2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~	11	~~12~~	13	~~14~~	~~15~~	~~16~~
2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~	11	~~12~~	13	~~14~~	~~15~~	~~16~~
2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~	11	~~12~~	13	~~14~~	~~15~~	~~16~~

Primes and Arithmetic Progressions

- Euclid proved that there are infinitely many primes.
- G. Lejuenne Dirchlet also showed that every arithmetic progression $ka + b$, $k = 1, 2, \ldots$, where a & b have no common factor greater than 1 contains infinitely many primes in the 19th century
- Are there long arithmetic progressions made up entirely of primes?

Primes and Arithmetic Progressions

- Euclid proved that there are infinitely many primes.
- G. Lejuenne Dirchlet also showed that every arithmetic progression $ka + b$, $k = 1, 2, \ldots$, where a & b have no common factor greater than 1 contains infinitely many primes in the 19th century
- Are there long arithmetic progressions made up entirely of primes?
  - 5,11, 17, 23, 29 is an arithmetic progression of **5 primes**.
  - 199, 409, 619, 829, 1039,1249, 1459, 1669, 1879, 2089 is an arithmetic progression of **10 primes**.
- In the 1930s, Paul Erdös conjectured that for every positive integer $n > 1$, there is an arithmetic progression of length n made up entirely of primes. This was proven in 2006, by Ben Green and Terence Tao.

Generating Primes

- Number theory is noted as a subject for which it is easy to formulate conjectures, some of which are difficult to prove and others that remained open problems for many years.
- It would be useful to have a function $f(n)$ s/t $f(n)$ is prime $\forall n \in \mathbb{N}$.

Generating Primes

- Number theory is noted as a subject for which it is easy to formulate conjectures, some of which are difficult to prove and others that remained open problems for many years.
- It would be useful to have a function $f(n)$ s/t $f(n)$ is prime $\forall n \in \mathbb{N}$.
- If we had such a function, we could generate large primes for use in cryptography and other applications.
- Consider the polynomial $f(n) = n^2 - n + 41$.

Generating Primes

- Number theory is noted as a subject for which it is easy to formulate conjectures, some of which are difficult to prove and others that remained open problems for many years.
- It would be useful to have a function $f(n)$ s/t $f(n)$ is prime $\forall n \in \mathbb{N}$.
- If we had such a function, we could generate large primes for use in cryptography and other applications.
- Consider the polynomial $f(n) = n^2 - n + 41$. This polynomial has the interesting property that $f(n)$ is prime for all positive integers $n \leq 40$.

Generating Primes

- The problem of generating large primes is of both theoretical and practical interest.
- Finding large primes, say with 600 hundred of digits, is important in cryptography.
- So far, no useful closed formula that always produces primes has been found.
- Fortunately, we can generate large integers which are almost certainly primes.

Generating Primes

- The problem of generating large primes is of both theoretical and practical interest.
- Finding large primes, say with 600 hundred of digits, is important in cryptography.
- So far, no useful closed formula that always produces primes has been found.
- Fortunately, we can generate large integers which are almost certainly primes.
- In 2002, AKS gave algorithm PRIMES is in $\mathcal{P}$
- Miller-Rabin primality test proposed in 1980. It's a probabilistic algorithm. It is normally used to check primality of large number.

Carmichael Numbers

Definition

A composite integer n *that satisfies the congruence* $b^{n-1} \equiv 1 \mod n \ \forall \ b, b \in \mathbb{N}$ *with* $\gcd(b, n) = 1$ *is called a Carmichael number.*

Carmichael Numbers

Definition

A composite integer n that satisfies the congruence $b^{n-1} \equiv 1 \mod n \;\forall\; b, b \in \mathbb{N}$ with $\gcd(b, n) = 1$ is called a Carmichael number.

Example

The integer 561 is a Carmichael number. To see this:

- $561 = 3 \times 11 \times 17.$

Carmichael Numbers

Definition

A composite integer n that satisfies the congruence $b^{n-1} \equiv 1 \mod n \; \forall \; b, b \in \mathbb{N}$ with $\gcd(b, n) = 1$ is called a Carmichael number.

Example

The integer 561 is a Carmichael number. To see this:

- $561 = 3 \times 11 \times 17$.
- If $\gcd(b, 561) = 1$, then $\gcd(b, 3) = 1$, $\gcd(b, 11) = 1$ and $\gcd(b, 17) = 1$.
- If $\gcd(b, 561) = 1$, we have

Carmichael Numbers

Definition

A composite integer n that satisfies the congruence $b^{n-1} \equiv 1 \mod n \ \forall\ b, b \in \mathbb{N}$ with $\gcd(b,n) = 1$ is called a Carmichael number.

Example

The integer 561 is a Carmichael number. To see this:

- $561 = 3 \times 11 \times 17.$
- If $\gcd(b,561) = 1$, then $\gcd(b,3) = 1$, $\gcd(b,11) = 1$ and $\gcd(b,17) = 1$.
- If $\gcd(b,561) = 1$, we have

$$b^{560} = \left(b^2\right)^{280} \equiv 1 \mod 3,$$

$$b^{560} = \left(b^{10}\right)^{56} \equiv 1 \mod 11,$$

$$b^{560} = \left(b^{16}\right)^{35} \equiv 1 \mod 17.$$

Carmichael Numbers

Definition

A composite integer n that satisfies the congruence $b^{n-1} \equiv 1 \mod n \ \forall\, b, b \in \mathbb{N}$ with $\gcd(b, n) = 1$ is called a Carmichael number.

Example

The integer 561 is a Carmichael number. To see this:

- $561 = 3 \times 11 \times 17$.
- If $\gcd(b, 561) = 1$, then $\gcd(b, 3) = 1$, $\gcd(b, 11) = 1$ and $\gcd(b, 17) = 1$.
- If $\gcd(b, 561) = 1$, we have

$$b^{560} = \left(b^2\right)^{280} \equiv 1 \mod 3,$$

$$b^{560} = \left(b^{10}\right)^{56} \equiv 1 \mod 11,$$

$$b^{560} = \left(b^{16}\right)^{35} \equiv 1 \mod 17.$$

- $\Rightarrow b^{560} \equiv 1 \mod 561$

Carmichael Numbers

Example

All Carmichael numbers < 10000:

(i) $561 = 3 \times 11 \times 17$
(ii) $1105 = 5 \times 13 \times 17$
(iii) $1729 = 7 \times 13 \times 19$
(iv) $2465 = 5 \times 17 \times 29$
(v) $2821 = 7 \times 13 \times 31$
(vi) $6601 = 7 \times 23 \times 41$
(vii) $8911 = 7 \times 19 \times 67$

Carmichael Numbers

Example

All Carmichael numbers < 10000:

(i) $561 = 3 \times 11 \times 17$

(ii) $1105 = 5 \times 13 \times 17$

(iii) $1729 = 7 \times 13 \times 19$

(iv) $2465 = 5 \times 17 \times 29$

(v) $2821 = 7 \times 13 \times 31$

(vi) $6601 = 7 \times 23 \times 41$

(vii) $8911 = 7 \times 19 \times 67$

- Carmichael number with 4 prime factors $62745 = 3 \times 5 \times 47 \times 89$

Carmichael Numbers

Example

All Carmichael numbers < 10000:

(i) $561 = 3 \times 11 \times 17$
(ii) $1105 = 5 \times 13 \times 17$
(iii) $1729 = 7 \times 13 \times 19$
(iv) $2465 = 5 \times 17 \times 29$
(v) $2821 = 7 \times 13 \times 31$
(vi) $6601 = 7 \times 23 \times 41$
(vii) $8911 = 7 \times 19 \times 67$

- Carmichael number with 4 prime factors $62745 = 3 \times 5 \times 47 \times 89$
- There are infinitely many Carmichael numbers

Carmichael Numbers

Theorem

Korselt's Criterion for Carmichael Numbers Let n be a composite number. Then n is a Carmichael number iff it is odd and every prime p dividing n satisfies the following two conditions:

(i) $p^2 \nmid n$

(ii) $(p-1) \mid (n-1)$

Quadratic Residue

Quadratic Residue

Example

b	1	2	3	4	5	6	7	8	9	10	11	12
b^2	1	4	9	3	12	10	10	12	3	9	4	1

$$\bmod 13$$

Quadratic Residue

Example

b	1	2	3	4	5	6	7	8	9	10	11	12
b^2	1	4	9	3	12	10	10	12	3	9	4	1

$$\bmod 13$$

- Is 3 congruent to the square of some number modulo 13?
- Does the congruence $x^2 \equiv -1 \mod 13$ have a solution?

Quadratic Residue

Example

b	1	2	3	4	5	6	7	8	9	10	11	12
b^2	1	4	9	3	12	10	10	12	3	9	4	1

$$\bmod 13$$

- Is 3 congruent to the square of some number modulo 13?
- Does the congruence $x^2 \equiv -1 \mod 13$ have a solution?

Definition

A nonzero number that is congruent to a square modulo p is called a quadratic residue $\bmod p$. A number that is not congruent to a square modulo p is called a quadratic nonresidue $\bmod p$.

Quadratic Residue

Definition

*Let $a \in \mathbb{Z}_n^*$; a is said to be a **quadratic residue** modulo n, if*

$$\exists\, x \in \mathbb{Z}_n^* \ni x^2 \equiv a \mod n.$$

*If no such x exists, then a is called a **quadratic non-residue** modulo n.*

The set of all quadratic residues modulo n is denoted by Q_n and the set of all quadratic non-residues is denoted by $\overline{Q_n}$.

Quadratic Residue

Definition

Let $a \in \mathbb{Z}_n^*$; a is said to be a ***quadratic residue*** *modulo n, if*

$$\exists\, x \in \mathbb{Z}_n^* \ni x^2 \equiv a \mod n.$$

If no such x exists, then a is called a ***quadratic non-residue*** *modulo n.*

The set of all quadratic residues modulo n is denoted by Q_n and the set of all quadratic non-residues is denoted by $\overline{Q_n}$.

- Let p be an odd prime and let α be a generator of $\mathbb{Z}_p^*$. Then $a \in \mathbb{Z}_p^*$ is a quadratic residue modulo $p \Leftrightarrow a \equiv \alpha^i \mod p$, where i is an even integer.

Quadratic Residue

Definition

*Let $a \in \mathbb{Z}_n^*$; a is said to be a **quadratic residue** modulo n, if*

$$\exists\ x \in \mathbb{Z}_n^* \ \ni\ x^2 \equiv a \mod n.$$

*If no such x exists, then a is called a **quadratic non-residue** modulo n.*

The set of all quadratic residues modulo n is denoted by Q_n and the set of all quadratic non-residues is denoted by $\overline{Q_n}$.

- Let p be an odd prime and let α be a generator of $\mathbb{Z}_p^*$. Then $a \in \mathbb{Z}_p^*$ is a quadratic residue modulo $p \Leftrightarrow a \equiv \alpha^i \mod p$, where i is an even integer.
- It follows that $\#Q_p = \frac{p-1}{2}$ and $\#\overline{Q_p} = \frac{p-1}{2}$.

Quadratic Residue

Definition

Let $a \in \mathbb{Z}_n^*$; a is said to be a ***quadratic residue*** *modulo n, if*

$$\exists\ x \in \mathbb{Z}_n^* \ \ni\ x^2 \equiv a \mod n.$$

If no such x exists, then a is called a ***quadratic non-residue*** *modulo n.*

The set of all quadratic residues modulo n is denoted by Q_n and the set of all quadratic non-residues is denoted by $\overline{Q_n}$.

- Let p be an odd prime and let α be a generator of $\mathbb{Z}_p^*$. Then $a \in \mathbb{Z}_p^*$ is a quadratic residue modulo $p \Leftrightarrow a \equiv \alpha^i \mod p$, where i is an even integer.
- It follows that $\#Q_p = \frac{p-1}{2}$ and $\#\overline{Q_p} = \frac{p-1}{2}$.

Theorem

Let p be an odd prime. Then there are exactly $\frac{p-1}{2}$ quadratic residues and exactly $\frac{p-1}{2}$ quadratic nonresidues $mod\ p$.

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Hence $Q_{13} = \{1, 3, 4, 9, 10, 12\}$ and $\overline{Q_{13}} = \{2, 5, 6, 7, 8, 11\}$.

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Hence $Q_{13} = \{1, 3, 4, 9, 10, 12\}$ and $\overline{Q_{13}} = \{2, 5, 6, 7, 8, 11\}$.

- Let $n = p.q$ be a product of two distinct odd primes. Then $a \in \mathbb{Z}_n^*$ is a quadratic residue modulo $n \Leftrightarrow a \in Q_p \;\&\; a \in Q_q$.

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Hence $Q_{13} = \{1, 3, 4, 9, 10, 12\}$ and $\overline{Q_{13}} = \{2, 5, 6, 7, 8, 11\}$.

- Let $n = p.q$ be a product of two distinct odd primes. Then $a \in \mathbb{Z}_n^*$ is a quadratic residue modulo $n \Leftrightarrow a \in Q_p \ \& \ a \in Q_q$.
- It follows that $\#Q_n = \frac{(p-1)(q-1)}{4}$ and $\#\overline{Q_n} = \frac{3(p-1)(q-1)}{4}$.

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Hence $Q_{13} = \{1, 3, 4, 9, 10, 12\}$ and $\overline{Q_{13}} = \{2, 5, 6, 7, 8, 11\}$.

- Let $n = p.q$ be a product of two distinct odd primes. Then $a \in \mathbb{Z}_n^*$ is a quadratic residue modulo $n \Leftrightarrow a \in Q_p \;\&\; a \in Q_q$.
- It follows that $\#Q_n = \frac{(p-1)(q-1)}{4}$ and $\#\overline{Q_n} = \frac{3(p-1)(q-1)}{4}$.

  Let $n = 21$.
  Then Q_{21}

Quadratic Residue

Example

$\alpha = 6$ is a generator of $\mathbb{Z}_{13}^*$. The powers of α are

i	0	1	2	3	4	5	6	7	8	9	10	11
$\alpha^i \mod 13$	1	6	10	8	9	2	12	7	3	5	4	11

Hence $Q_{13} = \{1, 3, 4, 9, 10, 12\}$ and $\overline{Q_{13}} = \{2, 5, 6, 7, 8, 11\}$.

- Let $n = p.q$ be a product of two distinct odd primes. Then $a \in \mathbb{Z}_n^*$ is a quadratic residue modulo $n \Leftrightarrow a \in Q_p \ \& \ a \in Q_q$.
- It follows that $\#Q_n = \frac{(p-1)(q-1)}{4}$ and $\#\overline{Q_n} = \frac{3(p-1)(q-1)}{4}$.

  Let $n = 21$.
  Then $Q_{21} = \{1, 4, 16\}$ and $\overline{Q_{21}} = \{2, 5, 8, 10, 11, 13, 17, 19, 20\}$.

The Legendre and Jacobi Symbols

- Let p be an odd prime and a an integer. The **Legendre symbol** $\left(\frac{a}{p}\right)$ is defined to be

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & \text{if } p \mid a, \\ 1, & \text{if } a \in Q_p, \\ -1, & \text{if } a \in \overline{Q_p}. \end{cases}$$

The Legendre and Jacobi Symbols

- Let p be an odd prime and a an integer. The **Legendre symbol** $\left(\frac{a}{p}\right)$ is defined to be

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & \text{if } p \mid a, \\ 1, & \text{if } a \in Q_p, \\ -1, & \text{if } a \in \overline{Q_p}. \end{cases}$$

- Let $n \geq 3$ be odd with prime factorization $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$. Then the **Jacobi symbol** $\left(\frac{a}{n}\right)$ is defined to be

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{e_1} \left(\frac{a}{p_2}\right)^{e_2} \cdots \left(\frac{a}{p_k}\right)^{e_k}$$

Properties of Legendre Symbol

(i) $\left(\frac{a}{p}\right) = a^{(p-1)/2} \mod p$. In particular, $\left(\frac{1}{p}\right) = 1$ and $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$. Hence, $-1 \in Q_p$ if $p \equiv 1 \mod 4$, and $-1 \in \overline{Q_p}$ if $p \equiv 3 \mod 4$.

Properties of Legendre Symbol

- (i) $\left(\frac{a}{p}\right) = a^{(p-1)/2} \mod p$. In particular, $\left(\frac{1}{p}\right) = 1$ and $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$. Hence, $-1 \in Q_p$ if $p \equiv 1 \mod 4$, and $-1 \in \overline{Q_p}$ if $p \equiv 3 \mod 4$.
- (ii) $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$. Hence if $a \in \mathbb{Z}_p^*$, then $\left(\frac{a^2}{p}\right) = 1$.

Properties of Legendre Symbol

- (i) $\left(\frac{a}{p}\right) = a^{(p-1)/2} \mod p$. In particular, $\left(\frac{1}{p}\right) = 1$ and $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$. Hence, $-1 \in Q_p$ if $p \equiv 1 \mod 4$, and $-1 \in \overline{Q_p}$ if $p \equiv 3 \mod 4$.
- (ii) $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$. Hence if $a \in \mathbb{Z}_p^*$, then $\left(\frac{a^2}{p}\right) = 1$.
- (iii) If $a \equiv b \mod p$, then $\left(\frac{a}{p}\right) = \left(\frac{b}{p}\right)$.

Properties of Legendre Symbol

(i) $\left(\frac{a}{p}\right) = a^{(p-1)/2} \mod p$. In particular, $\left(\frac{1}{p}\right) = 1$ and $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$. Hence, $-1 \in Q_p$ if $p \equiv 1 \mod 4$, and $-1 \in \overline{Q_p}$ if $p \equiv 3 \mod 4$.

(ii) $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$. Hence if $a \in \mathbb{Z}_p^*$, then $\left(\frac{a^2}{p}\right) = 1$.

(iii) If $a \equiv b \mod p$, then $\left(\frac{a}{p}\right) = \left(\frac{b}{p}\right)$.

(iv) **Law of quadratic reciprocity:** If q is an odd prime distinct from p, then

$$\left(\frac{p}{q}\right) = \left(\frac{q}{p}\right)(-1)^{(p-1)(q-1)/4}.$$

Properties of Legendre Symbol

Theorem (Law of Quadratic Reciprocity)

Let p and q be distinct odd primes.

$$\left(\frac{-1}{p}\right) = \begin{cases} 1, & \textit{if } p \equiv 1 \mod 4, \\ -1, & \textit{if } p \equiv 3 \mod 4, \end{cases}$$

Properties of Legendre Symbol

Theorem (Law of Quadratic Reciprocity)

Let p and q be distinct odd primes.

$$\left(\frac{-1}{p}\right) = \begin{cases} 1, & \text{if } p \equiv 1 \mod 4, \\ -1, & \text{if } p \equiv 3 \mod 4, \end{cases}$$

$$\left(\frac{2}{p}\right) = \begin{cases} 1, & \text{if } p \equiv 1 \text{ or } 7 \mod 8, \\ -1, & \text{if } p \equiv 3 \text{ or } 5 \mod 8, \end{cases}$$

Properties of Legendre Symbol

Theorem (Law of Quadratic Reciprocity)

Let p and q be distinct odd primes.

$$\left(\frac{-1}{p}\right) = \begin{cases} 1, & \text{if } p \equiv 1 \mod 4, \\ -1, & \text{if } p \equiv 3 \mod 4, \end{cases}$$

$$\left(\frac{2}{p}\right) = \begin{cases} 1, & \text{if } p \equiv 1 \text{ or } 7 \mod 8, \\ -1, & \text{if } p \equiv 3 \text{ or } 5 \mod 8, \end{cases}$$

$$\left(\frac{q}{p}\right) = \begin{cases} \left(\frac{p}{q}\right), & \text{if } p \equiv 1 \mod 4 \text{ or } q \equiv 1 \mod 4, \\ -\left(\frac{p}{q}\right), & \text{if } p \equiv q \equiv 3 \mod 4 \end{cases}$$

Examples

Example

$$\left(\frac{14}{137}\right) \quad =$$

Examples

Example

$$\left(\frac{14}{137}\right) = \left(\frac{2}{137}\right)\left(\frac{7}{137}\right)$$ Quadratic Residue Multiplication Rule

$$= \left(\frac{7}{137}\right)$$ Quadratic Reciprocity says $\left(\frac{2}{137}\right) = 1, \because \ 137 \equiv 1$

Examples

Example

$$
\begin{aligned}
\left(\frac{14}{137}\right) &= \left(\frac{2}{137}\right)\left(\frac{7}{137}\right) && \text{Quadratic Residue Multiplication Rule} \\
&= \left(\frac{7}{137}\right) && \text{Quadratic Reciprocity says } \left(\frac{2}{137}\right) = 1, \because\ 137 \equiv 1 \\
&= \left(\frac{137}{7}\right) && \text{Quadratic Reciprocity and } 137 \equiv 1 \pmod 4 \\
&= \left(\frac{4}{7}\right) && \text{reducing } 137 \pmod 7
\end{aligned}
$$

Examples

Example

$$\left(\frac{14}{137}\right) = \left(\frac{2}{137}\right)\left(\frac{7}{137}\right)$$ Quadratic Residue Multiplication Rule

$$= \left(\frac{7}{137}\right)$$ Quadratic Reciprocity says $\left(\frac{2}{137}\right) = 1, \because\ 137 \equiv 1$

$$= \left(\frac{137}{7}\right)$$ Quadratic Reciprocity and $137 \equiv 1 \mod 4$

$$= \left(\frac{4}{7}\right)$$ reducing $137 \mod 7$

$$= 1$$ $\because 4 = 2^2$ is certainly a square

Examples

Example

$$\begin{array}{rcll} \left(\frac{14}{137}\right) & = & \left(\frac{2}{137}\right)\left(\frac{7}{137}\right) & \text{Quadratic Residue Multiplication Rule} \\ & = & \left(\frac{7}{137}\right) & \text{Quadratic Reciprocity says } \left(\frac{2}{137}\right) = 1, \because\ 137 \equiv 1 \\ & = & \left(\frac{137}{7}\right) & \text{Quadratic Reciprocity and } 137 \equiv 1 \mod 4 \\ & = & \left(\frac{4}{7}\right) & \text{reducing } 137 \mod 7 \\ & = & 1 & \because 4 = 2^2 \text{ is certainly a square} \end{array}$$

Exercise

Compute

$$\left(\frac{55}{179}\right)$$

Generalized Law of Quadratic Reciprocity

Theorem (Generalized Law of Quadratic Reciprocity)

Let a and b be odd positive integers.

$$\left(\frac{-1}{b}\right) = \begin{cases} 1, & \text{if } b \equiv 1 \mod 4, \\ -1, & \text{if } b \equiv 3 \mod 4, \end{cases}$$

Generalized Law of Quadratic Reciprocity

Theorem (Generalized Law of Quadratic Reciprocity)

Let a and b be odd positive integers.

$$\left(\frac{-1}{b}\right) = \begin{cases} 1, & \textit{if } b \equiv 1 \mod 4, \\ -1, & \textit{if } b \equiv 3 \mod 4, \end{cases}$$

$$\left(\frac{2}{b}\right) = \begin{cases} 1, & \textit{if } b \equiv 1 \textit{ or } 7 \mod 8, \\ -1, & \textit{if } b \equiv 3 \textit{ or } 5 \mod 8, \end{cases}$$

Generalized Law of Quadratic Reciprocity

Theorem (Generalized Law of Quadratic Reciprocity)

Let a and b be odd positive integers.

$$\left(\frac{-1}{b}\right) = \begin{cases} 1, & \text{if } b \equiv 1 \mod 4, \\ -1, & \text{if } b \equiv 3 \mod 4, \end{cases}$$

$$\left(\frac{2}{b}\right) = \begin{cases} 1, & \text{if } b \equiv 1 \text{ or } 7 \mod 8, \\ -1, & \text{if } b \equiv 3 \text{ or } 5 \mod 8, \end{cases}$$

$$\left(\frac{a}{b}\right) = \begin{cases} \left(\frac{b}{a}\right), & \text{if } a \equiv 1 \mod 4 \text{ or } b \equiv 1 \mod 4, \\ -\left(\frac{b}{a}\right), & \text{if } a \equiv b \equiv 3 \mod 4 \end{cases}$$

Solovay-Strassen Theorem

Definition

If $n > 1$ is an odd integer then an integer $a \in \{1, \ldots, n-1\}$ s/t either

(i) $\gcd(a, n) > 1$, *or*

(ii) $\gcd(a, n) = 1$ *and* $a^{(n-1)/2} \not\equiv \left(\frac{a}{n}\right) \mod n$

is called an Euler witness for n.

Solovay-Strassen Theorem

Definition

If $n > 1$ is an odd integer then an integer $a \in \{1, \ldots, n-1\}$ s/t either

- (i) $\gcd(a, n) > 1$, *or*
- (ii) $\gcd(a, n) = 1$ *and* $a^{(n-1)/2} \not\equiv \left(\frac{a}{n}\right) \mod n$

is called an Euler witness for n.

Theorem

Let n be an odd composite positive integer. There is an integer $a \in \{1, \ldots, n-1\}$ s/t

$$\gcd(a, n) = 1 \text{ and } a^{(n-1)/2} \not\equiv \left(\frac{a}{n}\right) \mod n.$$

Property of Prime Numbers

Theorem

Let p *be an odd prime and write*

$$p - 1 = 2^k q \qquad \text{with } q \text{ odd.}$$

Let a *be any number not divisible by* p*. Then one of the following two conditions is true:*

(i) $a^q \equiv 1 \mod p$

(ii) *One of the numbers* $a^q, a^{2q}, a^{4q}, \ldots, a^{2^{k-1}q}$ *is congruent to* $-1 \mod p$.

Miller-Rabin Test for Composite Numbers

Theorem

Let n be an odd integer and write $n-1 = 2^k q$ with q odd. If both of the following conditions are true for some a not divisible by n, then n is a composite number

(i)

$$a^q \not\equiv 1 \mod n$$

(ii)

$$a^{2^i q} \not\equiv -1 \mod n, \qquad 0 \le i \le k-1$$

Miller-Rabin Test for Composite Numbers

- Let n be an odd integer and write $n - 1 = 2^k q$ with q odd.
- If n is prime and $1 \leq a \leq n - 1$ then $a^{n-1} - 1 \equiv 0 \mod n$

Miller-Rabin Test for Composite Numbers

- Let n be an odd integer and write $n - 1 = 2^k q$ with q odd.
- If n is prime and $1 \leq a \leq n - 1$ then $a^{n-1} - 1 \equiv 0 \mod n$

$$\begin{aligned} a^{2^k q} - 1 &= \left(a^{2^{k-1} q}\right)^2 - 1 \\ &= \left(a^{2^{k-1} q} - 1\right)\left(a^{2^{k-1} q} + 1\right) \\ &= \left(a^{2^{k-2} q} - 1\right)\left(a^{2^{k-2} q} + 1\right)\left(a^{2^{k-1} q} + 1\right) \end{aligned}$$

Miller-Rabin Test for Composite Numbers

- Let n be an odd integer and write $n-1 = 2^k q$ with q odd.
- If n is prime and $1 \leq a \leq n-1$ then $a^{n-1} - 1 \equiv 0 \mod n$

$$\begin{array}{rcl}
a^{2^k q} - 1 & = & \left(a^{2^{k-1} q}\right)^2 - 1 \\
& = & \left(a^{2^{k-1} q} - 1\right)\left(a^{2^{k-1} q} + 1\right) \\
& = & \left(a^{2^{k-2} q} - 1\right)\left(a^{2^{k-2} q} + 1\right)\left(a^{2^{k-1} q} + 1\right) \\
\vdots & \vdots & \vdots \\
& = & (a^q - 1)(a^q + 1)\left(a^{2q} + 1\right)\left(a^{4q} + 1\right)\ldots\left(a^{2^{k-1} q} + 1\right)
\end{array}$$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

$$2^{35} \equiv$$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

$$\begin{aligned} 2^{35} &\equiv 263 \mod 561, \\ 2^{2.35} &\equiv 263^2 \equiv \end{aligned}$$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

$$\begin{aligned} 2^{35} &\equiv 263 \mod 561, \\ 2^{2.35} &\equiv 263^2 \equiv 166 \mod 561, \\ 2^{4.35} &\equiv 166^2 \equiv \end{aligned}$$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

$$\begin{aligned}
2^{35} &\equiv 263 \mod 561, \\
2^{2.35} &\equiv 263^2 \equiv 166 \mod 561, \\
2^{4.35} &\equiv 166^2 \equiv 67 \mod 561, \\
2^{8.35} &\equiv 67^2 \equiv 1 \mod 561.
\end{aligned}$$

Miller-Rabin Test for Composite Numbers

Example

- We will apply the Miller-Rabin test for $n = 561$ with $a = 2$
- We have $n - 1 = 560 = 2^4 \times 35$

$$\begin{aligned} 2^{35} &\equiv 263 \mod 561, \\ 2^{2.35} &\equiv 263^2 \equiv 166 \mod 561, \\ 2^{4.35} &\equiv 166^2 \equiv 67 \mod 561, \\ 2^{8.35} &\equiv 67^2 \equiv 1 \mod 561. \end{aligned}$$

- Thus, 2 is a Miller-Rabin witness to the fact that 561 is a composite number.

Miller-Rabin Test for Composite Numbers

Exercise

Apply Miller-Rabin test for

1. $n = 13$
2. $n = 41$
3. $n = 30121$

Fermat Test for Primality – Probabilistic Algorithm

Fermat Test for Primality

Input: n
Output: YES if n is composite, NO otherwise.
Choose a random $b,\ 0 < b < n$
if $\gcd(b, n) > 1$ **then**
  return YES
end
else ;
if $b^{n-1} \not\equiv 1 \mod n$ **then**
  return YES
end
else ;
return NO

The Euler Test – Probabilistic Algorithm

- If n is an odd prime, we know that an integer can have at most two square roots, $\mod n$. In particular, the only square roots of $1 \mod n$ are ± 1.
- If $a \not\equiv 0 \mod n$, $a^{(n-1)/2}$ is a square root of $a^{n-1} \equiv 1 \mod n$, so $a^{(n-1)/2} \equiv \pm 1 \mod n$.

The Euler Test – Probabilistic Algorithm

- If n is an odd prime, we know that an integer can have at most two square roots, $\mod n$. In particular, the only square roots of $1 \mod n$ are ± 1.
- If $a \not\equiv 0 \mod n$, $a^{(n-1)/2}$ is a square root of $a^{n-1} \equiv 1 \mod n$, so $a^{(n-1)/2} \equiv \pm 1 \mod n$.
- If $a^{(n-1)/2} \not\equiv \pm 1 \mod n$ for some a with $a \not\equiv 0 \mod n$, then n is composite.

The Euler Test – Probabilistic Algorithm

- For a randomly chosen a with $a \not\equiv 0 \mod n$, compute $a^{(n-1)/2} \mod n$.

The Euler Test – Probabilistic Algorithm

- For a randomly chosen a with $a \not\equiv 0 \mod n$, compute $a^{(n-1)/2} \mod n$.

  (i) If $a^{(n-1)/2} \equiv \pm 1 \mod n$, declare n a **probable prime**, and optionally repeat the test a few more times.

The Euler Test – Probabilistic Algorithm

- For a randomly chosen a with $a \not\equiv 0 \mod n$, compute $a^{(n-1)/2} \mod n$.

  (i) If $a^{(n-1)/2} \equiv \pm 1 \mod n$, declare n a **probable prime**, and optionally repeat the test a few more times.

  If n is large and chosen at random, the probability that n is prime is very close to 1.

  (ii) If $a^{(n-1)/2} \not\equiv \pm 1 \mod n$, declare n **composite**.

  This is always correct.

The Euler Test – Probabilistic Algorithm

- For a randomly chosen a with $a \not\equiv 0 \mod n$, compute $a^{(n-1)/2} \mod n$.

  (i) If $a^{(n-1)/2} \equiv \pm 1 \mod n$, declare n a **probable prime**, and optionally repeat the test a few more times.

  If n is large and chosen at random, the probability that n is prime is very close to 1.

  (ii) If $a^{(n-1)/2} \not\equiv \pm 1 \mod n$, declare n **composite**.

  This is always correct.

The Euler test is more powerful than the Fermat test.

The Euler Test – Probabilistic Algorithm

The Euler test is more powerful than the Fermat test.

- If the Fermat test finds that n is composite, so does the Euler test.
- If n is an odd composite integer (other than a prime power), 1 has at least 4 square roots $\mod n$.
- So we can have $a^{(n-1)/2} \equiv \beta \mod n$, where $\beta \neq \pm 1$ is a square root of 1.

The Euler Test – Probabilistic Algorithm

The Euler test is more powerful than the Fermat test.

- If the Fermat test finds that n is composite, so does the Euler test.
- If n is an odd composite integer (other than a prime power), 1 has at least 4 square roots $\mod n$.
- So we can have $a^{(n-1)/2} \equiv \beta \mod n$, where $\beta \neq \pm 1$ is a square root of 1.
- Then $a^{n-1} \equiv 1 \mod n$. In this situation, the Fermat Test (incorrectly) declares n a probable prime, but the Euler test (correctly) declares n composite.

Miller-Rabin Test – Probabilistic Algorithm

- The Euler test improves upon the Fermat test by taking advantage of the fact, if 1 has a square root other than $\pm 1 \mod n$, then n must be composite.
- If $a^{(n-1)/2} \not\equiv \pm 1 \mod n$, where $\gcd(a, n) = 1$, then n must be composite for one of two reasons:
  - (i) If $a^{n-1} \not\equiv 1 \mod n$, then n must be composite by Fermat's Little Theorem
  - (ii) If $a^{n-1} \equiv 1 \mod n$, then n must be composite because $a^{(n-1)/2}$ is a square root of $1 \mod n$ different from ± 1.

Miller-Rabin Test – Probabilistic Algorithm

- The Euler test improves upon the Fermat test by taking advantage of the fact, if 1 has a square root other than $\pm 1 \mod n$, then n must be composite.
- If $a^{(n-1)/2} \not\equiv \pm 1 \mod n$, where $\gcd(a, n) = 1$, then n must be composite for one of two reasons:
  - (i) If $a^{n-1} \not\equiv 1 \mod n$, then n must be composite by Fermat's Little Theorem
  - (ii) If $a^{n-1} \equiv 1 \mod n$, then n must be composite because $a^{(n-1)/2}$ is a square root of $1 \mod n$ different from ± 1.
- The limitation of the Euler test is that is does not go to any special effort to find square roots of 1, different from ± 1. The Miller-Rabin test does this.

Miller-Rabin Test – Probabilistic Algorithm

Miller-Rabin Test

Input: an odd integer $n \geq 3$ and security parameter $t \geq 1$.
Output: an answer "prime" or "composite" to the question: "Is n prime?"
Write $n - 1 = 2^s.r$ s/t r is odd.

for $i = 1$ *to* t **do**
- Choose a random integer a s/t $2 \leq a \leq n - 2$.
- Compute $y \equiv a^r \mod n$
- **if** $y \neq 1$ & $y \neq n - 1$ **then**
  - $j \leftarrow 1$.
  - **while** $j \leq s - 1$ & $y \neq n - 1$ **do**
    - Compute $y \leftarrow y^2 \mod n$.
    - If $y = 1$ then **return**("composite").
    - $j \leftarrow j + 1$.
  - **end**
  - If $y \neq n - 1$ then **return** ("composite").
- **end**

end

Return("prime").

Miller-Rabin Test

- The Miller-Rabin test is very fast and easy to implement on a computer, since, after computing $a^r \mod n$, we simply compute a few squares $\mod n$.

Miller-Rabin Test

- The Miller-Rabin test is very fast and easy to implement on a computer, since, after computing $a^r \mod n$, we simply compute a few squares $\mod n$.
- If n is an odd composite number, then at least 75% of the numbers a between 1 and $n-1$ act as Miller-Rabin witnesses for n.

Miller-Rabin Test

- The Miller-Rabin test is very fast and easy to implement on a computer, since, after computing $a^r \mod n$, we simply compute a few squares $\mod n$.
- If n is an odd composite number, then at least 75% of the numbers a between 1 and $n - 1$ act as Miller-Rabin witnesses for n.
- If we randomly choose 100 different values for a, and if none of them are Miller-Rabin witnesses for n, then the probability of n being composite $< 2^{-200} \approx 6 \times 10^{-61}$.

Deterministic Polynomial Time Algorithm

Idea of The AKS Algorithm

- Let $a \in \mathbb{Z},\ n \in \mathbb{N},\ n \geq 2$, and $\gcd(a, n) = 1$. Then n is prime iff

$$(X + a)^n \equiv$$

Deterministic Polynomial Time Algorithm

Idea of The AKS Algorithm

- Let $a \in \mathbb{Z},\ n \in \mathbb{N},\ n \geq 2$, and $\gcd(a, n) = 1$. Then n is prime iff

$$(X + a)^n \equiv X^n + a \mod n.$$

Deterministic Polynomial Time Algorithm

Idea of The AKS Algorithm

- Let $a \in \mathbb{Z},\ n \in \mathbb{N},\ n \geq 2$, and $\gcd(a, n) = 1$. Then n is prime iff

$$(X + a)^n \equiv X^n + a \mod n.$$

- Test the following equation:

$$(X + a)^n \equiv X^n + a \ (mod(X^r - 1), n),$$

for an appropriately chosen small r.

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$
Output: n is **Prime** or **Composite** in deterministic polynomial-time

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$
Output: n is **Prime** or **Composite** in deterministic polynomial-time
If $n = a^b$ with $a \in \mathbb{N}$ & $b > 1$, then output **COMPOSITE.**

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$

Output: n is **Prime** or **Composite** in deterministic polynomial-time

If $n = a^b$ with $a \in \mathbb{N}$ & $b > 1$, then output **COMPOSITE.**

Find the smallest r such that $ord_r(n) > 4(\log n)^2$.

If $1 < \gcd(a, n) < n$ for some $a \leq r$, then output **COMPOSITE.**

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$
Output: n is **Prime** or **Composite** in deterministic polynomial-time
If $n = a^b$ with $a \in \mathbb{N}$ & $b > 1$, then output **COMPOSITE.**
Find the smallest r such that $ord_r(n) > 4(\log n)^2$.
If $1 < \gcd(a, n) < n$ for some $a \leq r$, then output **COMPOSITE.**
If $n \leq r$, then output **PRIME.**

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$
Output: n is **Prime** or **Composite** in deterministic polynomial-time
If $n = a^b$ with $a \in \mathbb{N}$ & $b > 1$, then output **COMPOSITE.**
Find the smallest r such that $ord_r(n) > 4(\log n)^2$.
If $1 < \gcd(a, n) < n$ for some $a \leq r$, then output **COMPOSITE.**
If $n \leq r$, then output **PRIME.**
for $a = 1$ *to* $\lfloor 2\sqrt{\phi(r)} \log n \rfloor$ **do**
    if $(x - a)^n \not\equiv (x^n - a) \mod (x^r - 1, n)$,
    then output **COMPOSITE.**
end

Return("PRIME").

Deterministic Polynomial Time Algorithm

The AKS Algorithm

Input: a positive integer $n > 1$
Output: n is **Prime** or **Composite** in deterministic polynomial-time
If $n = a^b$ with $a \in \mathbb{N}$ & $b > 1$, then output **COMPOSITE.**
Find the smallest r such that $ord_r(n) > 4(\log n)^2$.
If $1 < \gcd(a, n) < n$ for some $a \leq r$, then output **COMPOSITE.**
If $n \leq r$, then output **PRIME.**
for $a = 1$ *to* $\lfloor 2\sqrt{\phi(r)} \log n \rfloor$ **do**
  if $(x - a)^n \not\equiv (x^n - a) \mod (x^r - 1, n)$,
  then output **COMPOSITE.**
end
Return(“PRIME").

Time Complexity = $\mathcal{O}(\log^6 n)$

References

- Tom M. Apostol,
  Introduction to Analytical Number Theory, Springer, 1976.
- Owen D. Byer, Deirdre L. Smeltzer, and Kenneth L. Wantz,
  Journey into Discrete Mathematics, MAA Press, 2018.
- Gerard O'Regan,
  Guide to Discrete Mathematics: An Accessible Introduction to the History, Theory, Logic and Applications, Springer, 2016.
- Kenneth H. Rosen,
  Discrete Mathematics and Its Applications, McGraw-Hill, 2019.

The End

Thanks a lot for your attention!