Discrete Mathematics for Computer Science

Lecture 5: Set and Function

Dr. Ming Tang

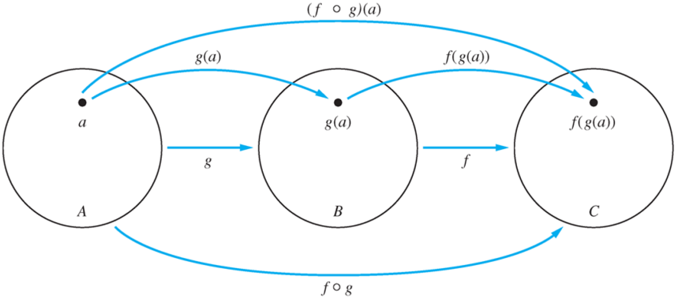
Department of Computer Science and Engineering
Southern University of Science and Technology (SUSTech)
Email: tangm3@sustech.edu.cn

Composition of Functions

Let f be a function from B to C and let g be a function from A to B. The composition of the functions f and g, denoted by $f \circ g$, is defined by $(f \circ g)(x) = f(g(x))$.

Composition of Functions

Let f be a function from B to C and let g be a function from A to B. The composition of the functions f and g, denoted by $f \circ g$, is defined by $(f \circ g)(x) = f(g(x))$.



Composition of Functions

- **Example 1**:

  Let $A = \{1, 2, 3\}$ and $B = \{a, b, c, d\}$.

$g : A \to A$	$f : A \to B$
$1 \mapsto 3$	$1 \mapsto b$
$2 \mapsto 1$	$2 \mapsto a$
$3 \mapsto 2$	$3 \mapsto d$

  What is $f \circ g$?

Composition of Functions

- **Example 1**:

  Let $A = \{1, 2, 3\}$ and $B = \{a, b, c, d\}$.

$g : A \to A$	$f : A \to B$
$1 \mapsto 3$	$1 \mapsto b$
$2 \mapsto 1$	$2 \mapsto a$
$3 \mapsto 2$	$3 \mapsto d$

  What is $f \circ g$?

  $f \circ g : A \to B$
  $1 \mapsto d$
  $2 \mapsto b$
  $3 \mapsto a$

Composition of Functions

- **Example 2**:

  Let $f : \mathbf{Z} \to \mathbf{Z}$ and $g : \mathbf{Z} \to \mathbf{Z}$, where $f(x) = 2x$ and $g(x) = x^2$.

  What are $g \circ f$ and $f \circ g$?

Composition of Functions

- **Example 2**:

  Let $f : \mathbf{Z} \rightarrow \mathbf{Z}$ and $g : \mathbf{Z} \rightarrow \mathbf{Z}$, where $f(x) = 2x$ and $g(x) = x^2$.

  What are $g \circ f$ and $f \circ g$?

$$g \circ f : \mathbf{Z} \rightarrow \mathbf{Z} \qquad g \circ f = 4x^2$$

$$f \circ g : \mathbf{Z} \rightarrow \mathbf{Z} \qquad f \circ g = 2x^2$$

Composition of Functions

- **Example 2**:

  Let $f : \mathbf{Z} \to \mathbf{Z}$ and $g : \mathbf{Z} \to \mathbf{Z}$, where $f(x) = 2x$ and $g(x) = x^2$.

  What are $g \circ f$ and $f \circ g$?

  $g \circ f : \mathbf{Z} \to \mathbf{Z} \qquad g \circ f = 4x^2$

  $f \circ g : \mathbf{Z} \to \mathbf{Z} \qquad f \circ g = 2x^2$

Note: In general, the order of composition matters.

SUSTech Southern University of Science and Technology

Composition of Functions

- Suppose that f is a bijection from A to B. Then $f \circ f^{-1} = I_B$ and $f^{-1} \circ f = I_A$, Since

$$(f^{-1} \circ f)(a) = f^{-1}(f(a)) = f^{-1}(b) = a$$
$$(f \circ f^{-1})(b) = f(f^{-1}(b)) = f(a) = b,$$

  where I_A, I_B denote the *identity functions* on the sets A and B, respectively.

Composition of Functions

- Suppose that f is a bijection from A to B. Then $f \circ f^{-1} = I_B$ and $f^{-1} \circ f = I_A$, Since
$$(f^{-1} \circ f)(a) = f^{-1}(f(a)) = f^{-1}(b) = a$$
$$(f \circ f^{-1})(b) = f(f^{-1}(b)) = f(a) = b,$$
where I_A, I_B denote the *identity functions* on the sets A and B, respectively.

Note: Identity function is sometimes denoted by $\iota_A(\cdot)$:

$$\iota_A(x) = x$$

Floor and Ceiling Functions

- The floor function assigns a real number x the largest integer that is $\leq x$, denoted by $\lfloor x \rfloor$. E.g., $\lfloor 3.5 \rfloor = 3$.
- The ceiling function assigns a real number x the smallest integer that is $\geq x$, denoted by $\lceil x \rceil$. E.g., $\lceil 3.5 \rceil = 4$.

Floor and Ceiling Functions

- The floor function assigns a real number x the largest integer that is $\leq x$, denoted by $\lfloor x \rfloor$. E.g., $\lfloor 3.5 \rfloor = 3$.
- The ceiling function assigns a real number x the smallest integer that is $\geq x$, denoted by $\lceil x \rceil$. E.g., $\lceil 3.5 \rceil = 4$.

(1a)	$\lfloor x \rfloor = n$ if and only if $n \leq x < n + 1$
(1b)	$\lceil x \rceil = n$ if and only if $n - 1 < x \leq n$
(1c)	$\lfloor x \rfloor = n$ if and only if $x - 1 < n \leq x$
(1d)	$\lceil x \rceil = n$ if and only if $x \leq n < x + 1$
(2)	$x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$
(3a)	$\lfloor -x \rfloor = -\lceil x \rceil$
(3b)	$\lceil -x \rceil = -\lfloor x \rfloor$
(4a)	$\lfloor x + n \rfloor = \lfloor x \rfloor + n$
(4b)	$\lceil x + n \rceil = \lceil x \rceil + n$

Note: n is an integer, x is a real number.

Floor and Ceiling Functions: Example 1

Prove that if x is a real number, then $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

Floor and Ceiling Functions: Example 1

Prove that if x is a real number, then $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

Proof: Let $x = n + \epsilon$, where n is an integer and $0 \leq \epsilon < 1$.

Floor and Ceiling Functions: Example 1

Prove that if x is a real number, then $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

Proof: Let $x = n + \epsilon$, where n is an integer and $0 \leq \epsilon < 1$.

- $0 \leq \epsilon < \frac{1}{2}$: In this case, $2x = 2n + 2\epsilon$. Since $0 \leq 2\epsilon < 1$, we have $\lfloor 2x \rfloor = 2n$. Similarly, $x + \frac{1}{2} = n + \frac{1}{2} + \epsilon$. Since $0 \leq \frac{1}{2} + \epsilon < 1$, we have $\lfloor x + \frac{1}{2} \rfloor = n$. Thus, $\lfloor 2x \rfloor = 2n$, and $\lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor = 2n$.

Floor and Ceiling Functions: Example 1

Prove that if x is a real number, then $\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor$.

Proof: Let $x = n + \epsilon$, where n is an integer and $0 \le \epsilon < 1$.

- $0 \le \epsilon < \frac{1}{2}$: In this case, $2x = 2n + 2\epsilon$. Since $0 \le 2\epsilon < 1$, we have $\lfloor 2x \rfloor = 2n$. Similarly, $x + \frac{1}{2} = n + \frac{1}{2} + \epsilon$. Since $0 \le \frac{1}{2} + \epsilon < 1$, we have $\lfloor x + \frac{1}{2} \rfloor = n$. Thus, $\lfloor 2x \rfloor = 2n$, and $\lfloor x \rfloor + \lfloor x + \frac{1}{2} \rfloor = 2n$.
- $\frac{1}{2} \le \epsilon < 1$: In this case, $2x = 2n + 2\epsilon = (2n+1) + (2\epsilon - 1)$. Since $0 \le 2\epsilon - 1 < 1$, we have $\lfloor 2x \rfloor = 2n + 1$. ....

Floor and Ceiling Functions: Example 2

Prove or disprove that $\lceil x + y \rceil = \lceil x \rceil + \lceil y \rceil$ for all real numbers x and y.

Floor and Ceiling Functions: Example 2

Prove or disprove that $\lceil x + y \rceil = \lceil x \rceil + \lceil y \rceil$ for all real numbers x and y.

Proof: This statement is false. Consider a counterexample $x = \frac{1}{2}$ and $y = \frac{1}{2}$. We can find that $\lceil x + y \rceil = 1$, but $\lceil x \rceil + \lceil y \rceil = 2$.

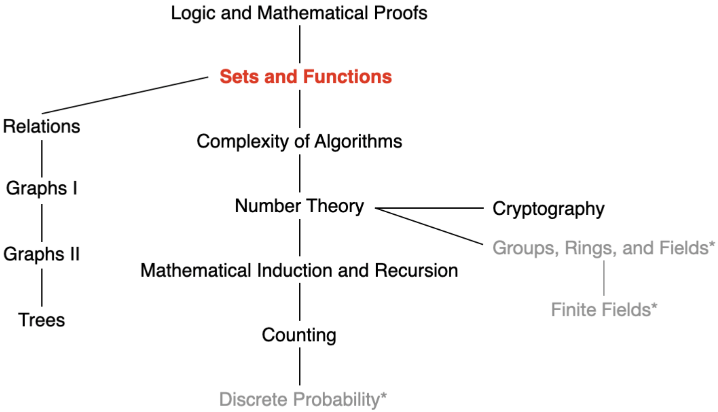
Factorial Function

The factorial function $f : \mathbf{N} \to \mathbf{Z}^+$ is the product of the first n positive integers when n is a nonnegative integer, denoted by $f(n) = n!$.

Summary of Function

- Function $f : A \to B$: an assignment of exactly one element of B to each element of A
- One-to-one function
- Onto function
- One-to-one correspondence: one-to-one function and onto
- Inverse function
- Floor function, ceiling function, factorial function

This Lecture



Set and Functions: set, set operations, functions, sequences and summation, cardinality of sets

Sequences

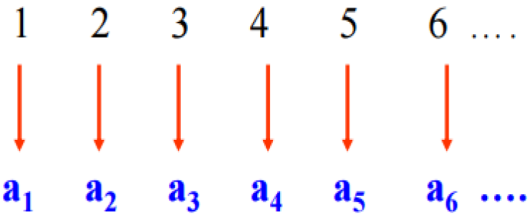
A sequence is a function from a subset of the set of integers (typically the set $\{0, 1, 2, \ldots\}$ or $\{1, 2, 3, \ldots\}$) to a set S.

We use the notation a_n to denote the image of the integer n. $\{a_n\}$ represents the ordered list $\{a_1, a_2, a_3, \ldots\}$

Sequences

A sequence is a function from a subset of the set of integers (typically the set $\{0, 1, 2, ...\}$ or $\{1, 2, 3, ...\}$) to a set S.

We use the notation a_n to denote the image of the integer n. $\{a_n\}$ represents the ordered list $\{a_1, a_2, a_3, ...\}$



Sequences

Examples:

- $a_n = n^2$, where $n = 1, 2, 3, \ldots$
- $a_n = (-1)^n$, where $n = 1, 2, 3, \ldots$
- $a_n = 2^n$, where $n = 1, 2, 3, \ldots$

Geometric Progression

A geometric progression is a sequence of the form

$$a, ar, ar^2, ..., ar^n, ...$$

where the initial term a and the common ratio r are real numbers.

Example: $a_n = 3 \times (\frac{1}{2})^n$, where $n = 0, 1, 2, 3, ...$

Arithmetic Progression

An arithmetic progression is a sequence of the form

$$a, a+d, a+2d, a+3d, ..., a+nd, ...$$

where the initial term a and common difference d are real numbers.

Example: $a_n = -1 + 4n$, where $n = 0, 1, 2, 3, ...$

Recursively Defined Sequences

1 Providing explicit formulas, e.g., $a_n = -1 + 4n$, where $n = 0, 1, 2, 3, \ldots$

Recursively Defined Sequences

1 Providing explicit formulas, e.g., $a_n = -1 + 4n$, where $n = 0, 1, 2, 3, \ldots$

2 Recursively Defined Sequences: provide

- one or more initial terms
- a rule for determining subsequent terms from those that precede them.

The n-th element of the sequence $\{a_n\}$ is defined recursively in terms of the previous elements of the sequence and the initial elements of the sequence.

Recursively Defined Sequences

1 Providing explicit formulas, e.g., $a_n = -1 + 4n$, where $n = 0, 1, 2, 3, \ldots$

2 Recursively Defined Sequences: provide

- one or more initial terms
- a rule for determining subsequent terms from those that precede them.

The n-th element of the sequence $\{a_n\}$ is defined recursively in terms of the previous elements of the sequence and the initial elements of the sequence.

Examples:

- $a_0 = 1$, $a_n = a_{n-1} + 2$ for $n = 1, 2, 3, \ldots.$
- $f_0 = 0$, $f_1 = 1$, $f_n = f_{n-1} + f_{n-2}$ for $n = 2, 3, 4, \ldots$ (Fibonacci sequence)

Summations

The summation of the terms of a sequence is

$$\sum_{j=m}^{n} a_j = a_m + a_{m+1} + \ldots + a_n$$

- j: the index of summation; the choice of the letter is arbitrary
- m: the lower limit of the summation
- n: the upper limit of the summation

Summations

The summation of the terms of a sequence is

$$\sum_{j=m}^{n} a_j = a_m + a_{m+1} + \ldots + a_n$$

- j: the index of summation; the choice of the letter is arbitrary
- m: the lower limit of the summation
- n: the upper limit of the summation

$$\sum_{j=1}^{n}(ax_j + by_j) = a\sum_{j=1}^{n} x_j + b\sum_{j=1}^{n} y_j$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} a_i b_j = \sum_{i=1}^{m} a_i \sum_{j=1}^{n} b_j$$

Summations

The sum of the first n terms of the arithmetic progression:

$$S_n = \sum_{j=0}^{n} (a + jd) = (n+1)a + d \sum_{j=0}^{n} j = (n+1)a + d\frac{n(n+1)}{2}$$

The sum of the first n terms of the geometric progression:

- $r \neq 1$

$$S_n = \sum_{j=0}^{n} (ar^j) = a \sum_{j=0}^{n} r^j = \frac{ar^{n+1} - a}{r - 1}$$

- $r = 1$

$$S_n = \sum_{j=0}^{n} (ar^j) = (n+1)a$$

Summations: Example

- **Examples**:
  - $S = \sum_{j=1}^{5}(2+3j)$
  - $S = \sum_{j=3}^{5}(2+3j)$
  - $S = \sum_{i=1}^{4}\sum_{j=1}^{2}(2i-j)$
  - $S = \sum_{j=0}^{3} 2(5)^j$
  - $S = \sum_{i=1}^{4}\sum_{j=1}^{3} ij$

Summations: Example

- **Examples**:
  - $S = \sum_{j=1}^{5}(2+3j)$ 55
  - $S = \sum_{j=3}^{5}(2+3j)$ 42
  - $S = \sum_{i=1}^{4}\sum_{j=1}^{2}(2i-j)$ 28
  - $S = \sum_{j=0}^{3} 2(5)^j$ 312
  - $S = \sum_{i=1}^{4}\sum_{j=1}^{3} ij$ 60

Infinite Series

Infinite geometric series can be computed in the closed form for $|x| < 1$.

$$\sum_{k=0}^{\infty} x^k = \lim_{n\to\infty} \sum_{k=0}^{n} x^k = \lim_{n\to\infty} \frac{x^{n+1}-1}{x-1} = \frac{1}{1-x}$$

Infinite Series

Infinite geometric series can be computed in the closed form for $|x| < 1$.

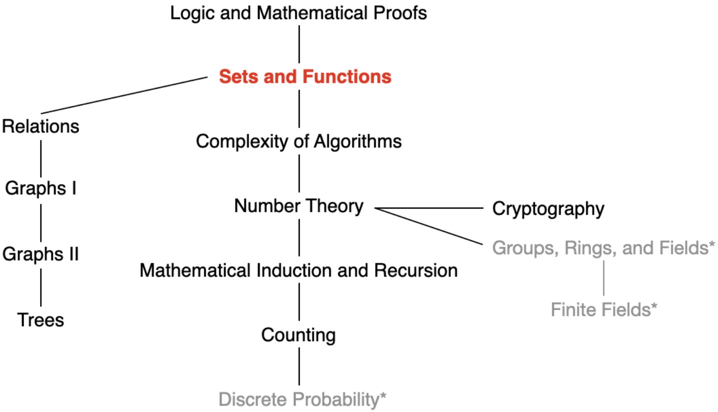
$$\sum_{k=0}^{\infty} x^k = \lim_{n\to\infty} \sum_{k=0}^{n} x^k = \lim_{n\to\infty} \frac{x^{n+1}-1}{x-1} = \frac{1}{1-x}$$

$$\sum_{k=0}^{\infty} kx^{k-1} = \frac{1}{(1-x)^2}$$

Some Useful Summation Formulas

$\sum_{k=0}^{n} ar^k \ (r \neq 0)$	$\frac{ar^{n+1} - a}{r - 1}, r \neq 1$
$\sum_{k=1}^{n} k$	$\frac{n(n+1)}{2}$
$\sum_{k=1}^{n} k^2$	$\frac{n(n+1)(2n+1)}{6}$
$\sum_{k=1}^{n} k^3$	$\frac{n^2(n+1)^2}{4}$
$\sum_{k=0}^{\infty} x^k, \lvert x\rvert < 1$	$\frac{1}{1-x}$
$\sum_{k=1}^{\infty} kx^{k-1}, \lvert x\rvert < 1$	$\frac{1}{(1-x)^2}$

This Lecture



Set and Functions: set, set operations, functions, sequences and summation, cardinality of sets

Cardinality of Sets

Recall: the cardinality of a finite set is defined by the number of the elements in the set.

Cardinality of Sets

Recall: the cardinality of a finite set is defined by the number of the elements in the set.

The sets A and B have the same cardinality if there is a one-to-one correspondence between elements in A and B.

Cardinality of Sets

Recall: the cardinality of a finite set is defined by the number of the elements in the set.

The sets A and B have the same cardinality if there is a one-to-one correspondence between elements in A and B.

If there is a one-to-one function from A to B, the cardinality of A is less than or equal to the cardinality of B, denoted by $|A| \leq |B|$.

Cardinality of Sets

Recall: the cardinality of a finite set is defined by the number of the elements in the set.

The sets A and B have the same cardinality if there is a one-to-one correspondence between elements in A and B.

If there is a one-to-one function from A to B, the cardinality of A is less than or equal to the cardinality of B, denoted by $|A| \leq |B|$.

Moreover, when $|A| \leq |B|$ and A and B have different cardinalities, we say that the cardinality of A is less than the cardinality of B, denoted by $|A| < |B|$.

Countable Sets

A set that is either finite or has the same cardinality as the set of positive integers $\mathbf{Z}^+$ is called countable. A set that is not countable is called uncountable.

Countable Sets

A set that is either finite or has the same cardinality as the set of positive integers $\mathbf{Z}^+$ is called countable. A set that is not countable is called uncountable.

Why are these called countable?

Countable Sets

A set that is either finite or has the same cardinality as the set of positive integers $\mathbf{Z}^+$ is called countable. A set that is not countable is called uncountable.

Why are these called countable?

The elements of the set can be enumerated and listed.

Hilbert's Paradox: Grand Hotel

The Grand Hotel has countably infinite number of rooms, each occupied by a guest. We can always accommodate a new guest at this hotel.

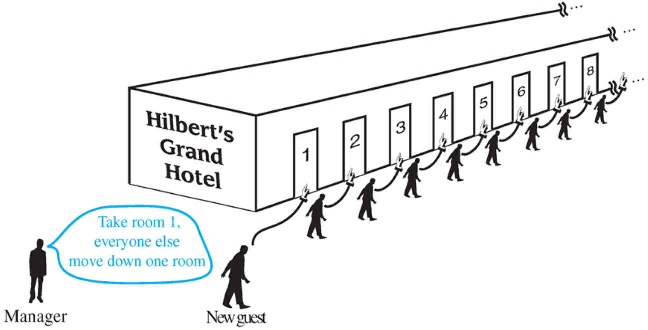


FIGURE 2 **A New Guest Arrives at Hilbert's Grand Hotel.**

Hilbert's Paradox: Grand Hotel

The Grand Hotel has countably infinite number of rooms, each occupied by a guest. We can always accommodate a new guest at this hotel.

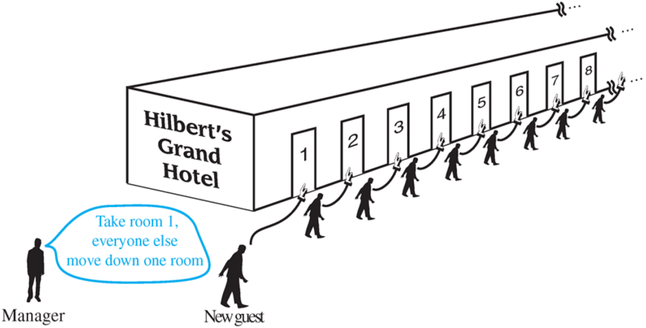


FIGURE 2 A New Guest Arrives at Hilbert's Grand Hotel.

Finitely many room: "All rooms are occupied" is equivalent to "no new guests can be accommodated".
Infinitely many room: This equivalence no longer holds.

Countable Sets: Example 1

The set of odd positive integers: $A = \{1, 3, 5, 7, ...\}$. Is it countable?

Countable Sets: Example 1

The set of odd positive integers: $A = \{1, 3, 5, 7, ...\}$. Is it countable?

Proof: Using the definition: If there is a one-to-one correspondence from the set of positive integers $\mathbf{Z}^+$ to this set A?

Countable Sets: Example 1

The set of odd positive integers: $A = \{1, 3, 5, 7, ...\}$. Is it countable?

Proof: Using the definition: If there is a one-to-one correspondence from the set of positive integers $\mathbf{Z}^+$ to this set A?

Consider the function

$$f(n) = 2n - 1$$

Countable Sets: Example 1

The set of odd positive integers: $A = \{1, 3, 5, 7, ...\}$. Is it countable?

Proof: Using the definition: If there is a one-to-one correspondence from the set of positive integers $\mathbf{Z}^+$ to this set A?

Consider the function

$$f(n) = 2n - 1$$

- One-to-one: Suppose $f(n) = f(m)$. Then, $2n - 1 = 2m - 1$, which leads to $n = m$.

Countable Sets: Example 1

The set of odd positive integers: $A = \{1, 3, 5, 7, ...\}$. Is it countable?

Proof: Using the definition: If there is a one-to-one correspondence from the set of positive integers $\mathbf{Z}^+$ to this set A?

Consider the function

$$f(n) = 2n - 1$$

- One-to-one: Suppose $f(n) = f(m)$. Then, $2n - 1 = 2m - 1$, which leads to $n = m$.
- Onto: For any arbitrary element in $t \in A$, we have an $n = (t + 1)/2 \in \mathbf{Z}^+$ such that $f(n) = t$.

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \dots$$

Thus, it is countable.

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \ldots$$

Thus, it is countable.

Theorem: An infinite set is countable if and only if it is possible to list the elements of the set in a sequence (indexed by the positive integers):

- All elements are listed

Why?

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \dots$$

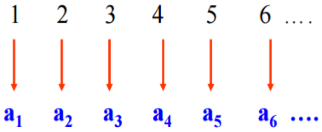
Thus, it is countable.

Theorem: An infinite set is countable if and only if it is possible to list the elements of the set in a sequence (indexed by the positive integers):

- All elements are listed

Why?

A sequence is a function from a subset of the set of integers to a set S.



Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \ldots$$

Thus, it is countable.

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \ldots$$

Thus, it is countable.

Alternatively, show there is a one-to-one correspondence from $\mathbf{Z}^+$ to $\mathbf{Z}$:

- when n is even: $f(n) = n/2$
- when n is odd: $f(n) = -(n-1)/2$

Thus, it is countable.

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, ...$$

Thus, it is countable.

Alternatively, show there is a one-to-one correspondence from $\mathbf{Z}^+$ to $\mathbf{Z}$:

- when n is even: $f(n) = n/2$
- when n is odd: $f(n) = -(n-1)/2$

Thus, it is countable.

Do $\mathbf{Z}^+$ and $\mathbf{Z}$ have the same cardinality?

Countable Sets: Example 2

Theorem: The set of integers $\mathbf{Z}$ is countable.

Proof: We can list the set of integers into a sequence:

$$0, 1, -1, 2, -2, 3, -3, \ldots$$

Thus, it is countable.

Alternatively, show there is a one-to-one correspondence from $\mathbf{Z}^+$ to $\mathbf{Z}$:

- when n is even: $f(n) = n/2$
- when n is odd: $f(n) = -(n-1)/2$

Thus, it is countable.

Do $\mathbf{Z}^+$ and $\mathbf{Z}$ have the same cardinality? Yes, because there is a one-to-one correspondence between $\mathbf{Z}^+$ and $\mathbf{Z}$.

Hilbert's Paradox: Grand Hotel

Countable Sets: Example 3

Theorem: The set of positive rational numbers is countable.

Countable Sets: Example 3

Theorem: The set of positive rational numbers is countable.

Hint: prove by showing that the set of positive rational numbers can be listed in a sequence: specifying the initial term and rule

Countable Sets: Example 3

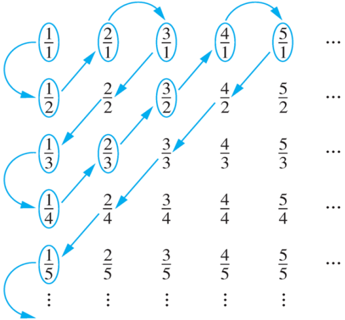
Theorem: The set of positive rational numbers is countable.

Hint: prove by showing that the set of positive rational numbers can be listed in a sequence: specifying the initial term and rule

Solution:

Constructing the list: first list p/q with $p+q=2$, next list p/q with $p+q=3$, and so on.

$1, 1/2, 2, 3, 1/3, 1/4, 2/3, \ldots$



SUSTech Southern University of Science and Technology

Countable Sets: Example 4

Theorem: The set of finite strings S over a finite alphabet A is countably infinite. (Assume an alphabetical ordering of symbols in A)

For example, let $A = \{$'a', 'b', 'c'$\}$. Then, set
$S = \{$'', 'a', 'b', 'c', 'ab' ..., 'aaaaa', ...$\}$

Countable Sets: Example 4

Theorem: The set of finite strings S over a finite alphabet A is countably infinite. (Assume an alphabetical ordering of symbols in A)

For example, let $A = \{$'a', 'b', 'c'$\}$. Then, set
$S = \{$'', 'a', 'b', 'c', 'ab' ..., 'aaaaa', ...$\}$

Solution:

We show that the strings can be listed in a sequence. First list

(i) all the strings of length 0 in alphabetical order.

(ii) then all the strings of length 1 in lexicographic order.

(iii) and so on.

This implies a bijection from $\mathbf{Z}^+$ to S.

Countable Sets: Example 5

The set of all Java programs is countable.

Countable Sets: Example 5

The set of all Java programs is countable.

Solution:

Let S be the set of strings constructed from the characters which may appear in a Java program. Use the ordering from the previous example. Take each string in turn

– feed the string into a Java compiler

– if the complier says YES, this is a syntactically correct Java program, we add this program to the list

– we move on to the next string

In this way, we construct a bijection from $\mathbf{Z}^+$ to the set of Java programs.

Countable Sets: Example 6

Theorem: Any subset of a countable set is countable.

Countable Sets: Example 6

Theorem: Any subset of a countable set is countable.

Proof: Consider a countable set A and its subset $B \subseteq A$.

Countable Sets: Example 6

Theorem: Any subset of a countable set is countable.

Proof: Consider a countable set A and its subset $B \subseteq A$.

- A is a finite set: $|B| \leq |A| < \infty$. Thus, $|B|$ is a finite set and hence countable.

Countable Sets: Example 6

Theorem: Any subset of a countable set is countable.

Proof: Consider a countable set A and its subset $B \subseteq A$.

- A is a finite set: $|B| \leq |A| < \infty$. Thus, $|B|$ is a finite set and hence countable.
- A is not a finite set: Since A is countable, the elements of A can be listed in a sequence. By removing the elements in the list that are not in B, we can obtain a list for B. Thus, B is countable

Countable Sets: Example 6

Theorem: Any subset of a countable set is countable.

Proof: Consider a countable set A and its subset $B \subseteq A$.

- A is a finite set: $|B| \leq |A| < \infty$. Thus, $|B|$ is a finite set and hence countable.
- A is not a finite set: Since A is countable, the elements of A can be listed in a sequence. By removing the elements in the list that are not in B, we can obtain a list for B. Thus, B is countable

Theorem: If A and B are countable sets, then $A \cup B$ is also countable.

Uncountable Sets: Example 1

A set that is not countable is called uncountable.

Theorem: The set of real numbers $\mathbf{R}$ is uncountable.

Proof by Contradiction:

Uncountable Sets: Example 1

A set that is not countable is called uncountable.

Theorem: The set of real numbers $\mathbf{R}$ is uncountable.

Proof by Contradiction: Suppose $\mathbf{R}$ is countable. Then, the interval from 0 to 1 is countable. This implies that the elements of this set can be listed as $r_1, r_2, r_3, \ldots$, where

- $r_1 = 0.d_{11}d_{12}d_{13}d_{14}$
- $r_2 = 0.d_{21}d_{22}d_{23}d_{24}$
- $r_3 = 0.d_{31}d_{32}d_{33}d_{34}$

where all $d_{ij} \in \{0, 1, 2, \ldots, 9\}$.

Uncountable Sets: Example 1

A set that is not countable is called uncountable.

Theorem: The set of real numbers $\mathbf{R}$ is uncountable.

Proof by Contradiction:
We want to show that not all real numbers in the interval between 0 and 1 are in this list. Form a new number called $r = 0.d_1d_2d_3d_4$, where $d_i = 2$ if $d_{ii} \neq 2$, and $d_i = 3$ if $d_{ii} = 2$.

Example: suppose	r1 = 0.75243...	d1 = 2
	r2 = 0.524310...	d2 = 3
	r3 = 0.131257...	d3 = 2
	r4 = 0.9363633...	d4 = 2
	...	...
	rt = 0.23222222...	dt = 3

Uncountable Sets: Example 1

A set that is not countable is called uncountable.

Theorem: The set of real numbers $\mathbf{R}$ is uncountable.

Proof by Contradiction:
We want to show that not all real numbers in the interval between 0 and 1 are in this list. Form a new number called $r = 0.d_1d_2d_3d_4$, where $d_i = 2$ if $d_{ii} \neq 2$, and $d_i = 3$ if $d_{ii} = 2$.

Example: suppose	r1 = 0.75243...	d1 = 2
	r2 = 0.524310...	d2 = 3
	r3 = 0.131257...	d3 = 2
	r4 = 0.9363633...	d4 = 2
	...	...
	rt = 0.23222222...	dt = 3

r and r_i differ in the i-th decimal place for all i.

Uncountable Sets: Example 1

A set that is not countable is called uncountable.

Theorem: The set of real numbers $\mathbf{R}$ is uncountable.

Proof by Contradiction:
We want to show that not all real numbers in the interval between 0 and 1 are in this list. Form a new number called $r = 0.d_1 d_2 d_3 d_4$, where $d_i = 2$ if $d_{ii} \neq 2$, and $d_i = 3$ if $d_{ii} = 2$.

Example: suppose	r1 = 0.75243...	d1 = 2
	r2 = 0.524310...	d2 = 3
	r3 = 0.131257...	d3 = 2
	r4 = 0.9363633...	d4 = 2
	...	...
	rt = 0.23222222...	dt = 3

r and r_i differ in the i-th decimal place for all i.
This leads to a contradiction.

Uncountable Sets: Example 2

Theorem: The set $\mathcal{P}(\mathbf{N})$ is uncountable.

Uncountable Sets: Example 2

Theorem: The set $\mathcal{P}(\mathbf{N})$ is uncountable.

Proof by contradiction:

Assume that $\mathcal{P}(\mathbb{N})$ is countable. This implies that the elements of this set can be listed as $S_0, S_1, S_2, \ldots$, where $S_i \subseteq \mathbb{N}$, and each S_i can be represented uniquely by the bit string $b_{i0}b_{i1}b_{i2}\ldots$, where $b_{ij} = 1$ if $j \in S_i$ and $b_{ij} = 0$ if $j \notin S_i$

- $S_0 = b_{00}b_{01}b_{02}b_{03}\cdots$
- $S_1 = b_{10}b_{11}b_{12}b_{13}\cdots$
- $S_2 = b_{20}b_{21}b_{22}b_{23}\cdots$

$\vdots$

all $b_{ij} \in \{0, 1\}$.

Uncountable Sets: Example 2

Theorem: The set $\mathcal{P}(\mathbf{N})$ is uncountable.

Proof by contradiction:

Assume that $\mathcal{P}(\mathbb{N})$ is countable. This implies that the elements of this set can be listed as $S_0, S_1, S_2, \ldots$, where $S_i \subseteq \mathbb{N}$, and each S_i can be represented uniquely by the bit string $b_{i0}b_{i1}b_{i2}\ldots$, where $b_{ij} = 1$ if $j \in S_i$ and $b_{ij} = 0$ if $j \notin S_i$

- $S_0 = b_{00}b_{01}b_{02}b_{03}\cdots$
- $S_1 = b_{10}b_{11}b_{12}b_{13}\cdots$
- $S_2 = b_{20}b_{21}b_{22}b_{23}\cdots$

$\vdots$

all $b_{ij} \in \{0, 1\}$.

Form a new set called $R = b_0b_1b_2b_3 \ldots$, where $b_i = 0$ if $b_{ii} = 1$, and $b_i = 1$ if $b_{ii} = 0$.

Uncountable Sets: Example 2

Theorem: The set $\mathcal{P}(\mathbf{N})$ is uncountable.

Proof by contradiction:

Assume that $\mathcal{P}(\mathbb{N})$ is countable. This implies that the elements of this set can be listed as $S_0, S_1, S_2, \ldots$, where $S_i \subseteq \mathbb{N}$, and each S_i can be represented uniquely by the bit string $b_{i0}b_{i1}b_{i2}\ldots$, where $b_{ij} = 1$ if $j \in S_i$ and $b_{ij} = 0$ if $j \notin S_i$

- $S_0 = b_{00}b_{01}b_{02}b_{03}\cdots$
- $S_1 = b_{10}b_{11}b_{12}b_{13}\cdots$
- $S_2 = b_{20}b_{21}b_{22}b_{23}\cdots$

$\vdots$

all $b_{ij} \in \{0, 1\}$.

Form a new set called $R = b_0b_1b_2b_3\ldots$, where $b_i = 0$ if $b_{ii} = 1$, and $b_i = 1$ if $b_{ii} = 0$. R is different from each set in the list. Each bit string is unique, and R and S_i differ in the i-th bit for all i.

Schroder-Bernstein Theorem

Theorem: If A and B are sets with $|A| \leq |B|$ and $|B| \leq |A|$, then $|A| = |B|$.

In other words, if there are one-to-one functions f from A to B and g from B to A, then there is a one-to-one correspondence between A and B, and hence $|A| = |B|$.

Schroder-Bernstein Theorem

Theorem: If A and B are sets with $|A| \leq |B|$ and $|B| \leq |A|$, then $|A| = |B|$.

In other words, if there are one-to-one functions f from A to B and g from B to A, then there is a one-to-one correspondence between A and B, and hence $|A| = |B|$.

Example: Show that $|(0,1)| = |(0,1]|$

Schroder-Bernstein Theorem

Theorem: If A and B are sets with $|A| \leq |B|$ and $|B| \leq |A|$, then $|A| = |B|$.

In other words, if there are one-to-one functions f from A to B and g from B to A, then there is a one-to-one correspondence between A and B, and hence $|A| = |B|$.

Example: Show that $|(0,1)| = |(0,1]|$

$f(x) = x$, $g(x) = x/2$

Computable vs Uncomputable

Definition: We say that a function is computable if there is a computer program in some programming language that finds the values of this function. If a function is not computable, we say it is uncomputable.

Computable vs Uncomputable

Definition: We say that a function is computable if there is a computer program in some programming language that finds the values of this function. If a function is not computable, we say it is uncomputable.

Theorem: There are functions that are not computable.

- The set of all programs is countable.
- There are uncountably many different functions from a particular countably infinite set to itself (omitted).

Cantor's Theorem

Cantor's theorem: If S is a set, then $|S| < |P(S)|$.

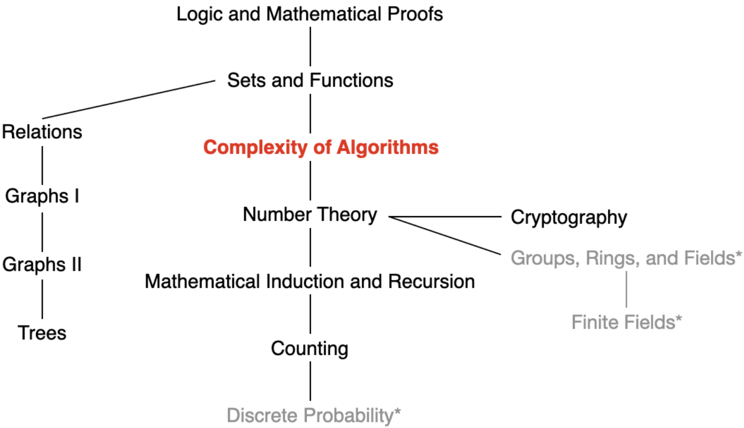
Prove by contradiction. Suppose there exists a function $f : S \to \mathcal{P}(S)$ that is surjective. Under this function f, we can define a set B such that

$$B = \{s \in S \mid s \notin f(S)\} \subseteq S.$$

Since f is surjective, the exists an b such that $f(b) = B$.

- If $b \in B$, then $b \in f(b)$: contradict to the definition of B;
- If $b \notin B$, then $b \notin f(b)$: $b \in B$, contradiction.

This Lecture



The growth of functions, complexity of algorithm, P and NP problem, ...

Algorithm

An algorithm is a finite sequence of <u>precise instructions</u> for performing a computation or for solving a problem.

Algorithm

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*$(a_1, \ldots, a_n$: real numbers with $n \geq 2)$
for $i := 1$ **to** $n - 1$
    for $j := 1$ **to** $n - i$
        if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
$\{a_1, \ldots, a_n$ is in increasing order$\}$

ALGORITHM 5 The Insertion Sort.

procedure *insertion sort*$(a_1, a_2, \ldots, a_n$: real numbers with $n \geq 2)$
for $j := 2$ **to** n
    $i := 1$
    while $a_j > a_i$
        $i := i + 1$
    $m := a_j$
    for $k := 0$ **to** $j - i - 1$
        $a_{j-k} := a_{j-k-1}$
    $a_i := m$
$\{a_1, \ldots, a_n$ is in increasing order$\}$

Algorithm

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*($a_1, \ldots, a_n$: real numbers with $n \geq 2$)
for $i := 1$ **to** $n - 1$
  for $j := 1$ **to** $n - i$
    if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
$\{a_1, \ldots, a_n$ is in increasing order$\}$

ALGORITHM 5 The Insertion Sort.

procedure *insertion sort*($a_1, a_2, \ldots, a_n$: real numbers with $n \geq 2$)
for $j := 2$ **to** n
  $i := 1$
  while $a_j > a_i$
    $i := i + 1$
  $m := a_j$
  for $k := 0$ **to** $j - i - 1$
    $a_{j-k} := a_{j-k-1}$
  $a_i := m$
$\{a_1, \ldots, a_n$ is in increasing order$\}$

How to compare algorithms with the same functionality?

Algorithm

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*($a_1, \ldots, a_n$: real numbers with $n \geq 2$)
for $i := 1$ **to** $n - 1$
  for $j := 1$ **to** $n - i$
    if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
$\{a_1, \ldots, a_n$ is in increasing order$\}$

ALGORITHM 5 The Insertion Sort.

procedure *insertion sort*($a_1, a_2, \ldots, a_n$: real numbers with $n \geq 2$)
for $j := 2$ **to** n
  $i := 1$
  while $a_j > a_i$
    $i := i + 1$
  $m := a_j$
  for $k := 0$ **to** $j - i - 1$
    $a_{j-k} := a_{j-k-1}$
  $a_i := m$
$\{a_1, \ldots, a_n$ is in increasing order$\}$

How to compare algorithms with the same functionality?

- Time complexity: number of machine operations (e.g., additions)

Algorithm

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*($a_1, \ldots, a_n$: real numbers with $n \geq 2$)
for $i := 1$ **to** $n - 1$
  for $j := 1$ **to** $n - i$
    if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
$\{a_1, \ldots, a_n$ is in increasing order$\}$

ALGORITHM 5 The Insertion Sort.

procedure *insertion sort*($a_1, a_2, \ldots, a_n$: real numbers with $n \geq 2$)
for $j := 2$ **to** n
  $i := 1$
  while $a_j > a_i$
    $i := i + 1$
  $m := a_j$
  for $k := 0$ **to** $j - i - 1$
    $a_{j-k} := a_{j-k-1}$
  $a_i := m$
$\{a_1, \ldots, a_n$ is in increasing order$\}$

How to compare algorithms with the same functionality?

- Time complexity: number of machine operations (e.g., additions)
- Space complexity: amount of memory needed

SUSTech Southern University of Science and Technology

Algorithm

An algorithm is a finite sequence of precise instructions for performing a computation or for solving a problem.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*($a_1, \ldots, a_n$: real numbers with $n \geq 2$)
for $i := 1$ **to** $n - 1$
  for $j := 1$ **to** $n - i$
    if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
$\{a_1, \ldots, a_n$ is in increasing order$\}$

ALGORITHM 5 The Insertion Sort.

procedure *insertion sort*($a_1, a_2, \ldots, a_n$: real numbers with $n \geq 2$)
for $j := 2$ **to** n
  $i := 1$
  while $a_j > a_i$
    $i := i + 1$
  $m := a_j$
  for $k := 0$ **to** $j - i - 1$
    $a_{j-k} := a_{j-k-1}$
  $a_i := m$
$\{a_1, \ldots, a_n$ is in increasing order$\}$

How to compare algorithms with the same functionality?

- Time complexity: number of machine operations (e.g., additions)
- Space complexity: amount of memory needed

Before we get into details, the growth of functions ...

The Growth of Functions

Which function is "bigger", $\frac{1}{10}n^2$ or $100n + 10000$?

The Growth of Functions

Which function is "bigger", $\frac{1}{10}n^2$ or $100n + 10000$?

It depends on the value of n.

The Growth of Functions

Which function is "bigger", $\frac{1}{10}n^2$ or $100n + 10000$?

It depends on the value of n.

In Computer Science, we are usually interested in what happens when our problem input size gets large.

The Growth of Functions

Which function is "bigger", $\frac{1}{10}n^2$ or $100n + 10000$?

It depends on the value of n.

In Computer Science, we are usually interested in what happens when our problem input size gets large.

Notice that when n is "large enough", $\frac{1}{10}n^2$ gets much bigger than $100n + 10000$ and stays larger.

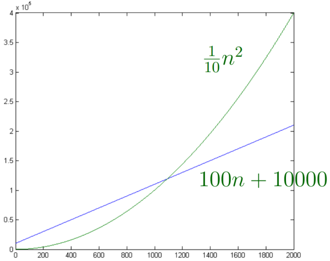
The Growth of Functions

Which function is "bigger", $\frac{1}{10}n^2$ or $100n + 10000$?

It depends on the value of n.

In Computer Science, we are usually interested in what happens when our problem input size gets large.

Notice that when n is "large enough", $\frac{1}{10}n^2$ gets much bigger than $100n + 10000$ and stays larger.



The Growth of Functions

- Big-O notation, e.g., $O(n^2)$
- Big-Omega notation, e.g., $\Omega(n^2)$
- Big-Theta notation, e.g., $\Theta(n^2)$

Big-O Notation

Let f and g be functions from the set of integers or the set of real numbers to the set of real numbers.We say that $f(x)$ is $O(g(x))$ if there are $\underline{\text{constants } C \text{ and } k}$ such that

$$|f(x)| \leq C|g(x)|,$$

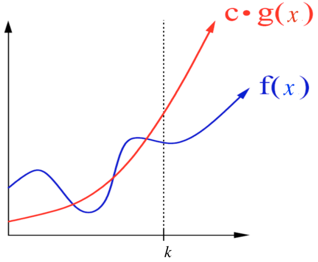
whenever $x > k$. [This is read as "$f(x)$ is big-oh of $g(x)$."]

Big-O Notation

Let f and g be functions from the set of integers or the set of real numbers to the set of real numbers.We say that $f(x)$ is $O(g(x))$ if there are constants C and k such that

$$|f(x)| \leq C|g(x)|,$$

whenever $x > k$. [This is read as "$f(x)$ is big-oh of $g(x)$."]



Big-O Notation: Example

Show that $f(x) = x^2 + 2x + 1$ is $O(x^2)$.

Big-O Notation: Example

Show that $f(x) = x^2 + 2x + 1$ is $O(x^2)$.

Proof: We can readily estimate the size of $f(x)$ when $x > 1$:

$$0 \leq x^2 + 2x + 1 \leq x^2 + 2x^2 + x^2 = 4x^2.$$

This is because when $x > 1$, $x < x^2$ and $1 < x^2$. Thus, let $C = 4$, $k = 1$:

$$|f(x)| \leq C|x^2|, \text{ whenever } x > k.$$

Hence, $f(x) = O(x^2)$.

SUSTech Southern University of Science and Technology

Big-O Notation: Example

Show that $f(x) = x^2 + 2x + 1$ is $O(x^2)$.

Proof: We can readily estimate the size of $f(x)$ when $x > 1$:

$$0 \leq x^2 + 2x + 1 \leq x^2 + 2x^2 + x^2 = 4x^2.$$

This is because when $x > 1$, $x < x^2$ and $1 < x^2$. Thus, let $C = 4$, $k = 1$:

$$|f(x)| \leq C|x^2|, \text{ whenever } x > k.$$

Hence, $f(x) = O(x^2)$.

Note that there are multiple ways for proving this. Alternatively, we can estimate the size of $f(x)$ when $x > 2$:

$$0 \leq x^2 + 2x + 1 \leq x^2 + x^2 + x^2 = 3x^2.$$

It follows that $C = 3$, $k = 2$. ...

Big-O Notation: Example

Examples: The following formulas are all $O(x^2)$:

- $4x^2$
- $8x^2 + 2x - 3$
- $x^2/5 + \sqrt{x} - \log(x)$

Big-O Notation: Example

Examples: The following formulas are all $O(x^2)$:

- $4x^2$
- $8x^2 + 2x - 3$
- $x^2/5 + \sqrt{x} - \log(x)$

Observe that in the relationship "$f(x)$ is $O(x^2)$," x^2 can be replaced by any function with "larger values" than x^2. For example,

- $f(x)$ is $O(x^3)$
- $f(x)$ is $O(x^2 + x + 7)$, ...

Big-O Notation: Example

Examples: The following formulas are all $O(x^2)$:

- $4x^2$
- $8x^2 + 2x - 3$
- $x^2/5 + \sqrt{x} - \log(x)$

Observe that in the relationship "$f(x)$ is $O(x^2)$," x^2 can be replaced by any function with "larger values" than x^2. For example,

- $f(x)$ is $O(x^3)$
- $f(x)$ is $O(x^2 + x + 7)$, ...

When $f(x)$ is $O(g(x))$, and $h(x)$ is a function that has larger absolute values than $g(x)$ does for sufficiently large values of x, it follows that

$$f(x) \text{ is } O(h(x)).$$

Big-O Estimates for Polynomials

Let $f(x) = a_n x^n + a_{n-1} x^{n-1} + ... + a_1 x + a_0$, where $a_0, a_1, ..., a_n$ are real numbers. Then, $f(x) = O(x^n)$.

Big-O Estimates for Polynomials

Let $f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0$, where $a_0, a_1, \ldots, a_n$ are real numbers. Then, $f(x) = O(x^n)$.

Proof:

Assuming $x > 1$, we have

$$\begin{aligned} |f(x)| &= |a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0| \\ &\leq |a_n| x^n + |a_{n-1}| x^{n-1} + \cdots + |a_1| x + |a_0| \\ &= x^n (|a_n| + |a_{n-1}|/x + \cdots + |a_1|/x^{n-1} + |a_0|/x^n) \\ &\leq x^n (|a_n| + |a_{n-1}| + \cdots + |a_1| + |a_0|). \end{aligned}$$

Big-O Estimates for Polynomials

Let $f(x) = a_n x^n + a_{n-1}x^{n-1} + ... + a_1 x + a_0$, where $a_0, a_1, ..., a_n$ are real numbers. Then, $f(x) = O(x^n)$.

Proof:

Assuming $x > 1$, we have

$$\begin{aligned} |f(x)| &= |a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0| \\ &\leq |a_n|x^n + |a_{n-1}|x^{n-1} + \cdots + |a_1|x + |a_0| \\ &= x^n(|a_n| + |a_{n-1}|/x + \cdots + |a_1|/x^{n-1} + |a_0|/x^n) \\ &\leq x^n(|a_n| + |a_{n-1}| + \cdots + |a_1| + |a_0|). \end{aligned}$$

The leading term $a_n x^n$ of a polynomial dominates its growth.

Big-O Estimates for Some Functions

$1 + 2 + \cdots + n = O(n^2)$

$n! = O(n^n)$

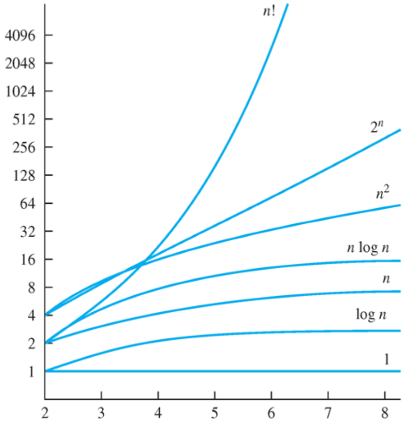
$\log n! = O(n \log n)$

$\log_a n = O(n)$ for an integer $a \geq 2$

$n^a = O(n^b)$ for integers $a \leq b$

$n^a = O(2^n)$ for an integer a

Big-O Estimates for Some Functions



Big-O Estimates for Some Functions

Prove $\log_a n = O(n)$ for an integer $a \geq 2$.

Big-O Estimates for Some Functions

Prove $\log_a n = O(n)$ for an integer $a \geq 2$.

Proof: We always have $\log_a n \leq n$ for $n > 1$. This can be proven using mathematical induction. ...

- $n = 1$: $log_a 1 = 0 < 1$
- Suppose $\log_a n \leq n$ for $n > 1$:

$$\log_a(n+1) \leq \log_a(an) = \log_a n + 1 \leq n + 1$$

Big-O Estimates for Some Functions

Prove $n^a = O(2^n)$ for an integer a.

Big-O Estimates for Some Functions

Prove $n^a = O(2^n)$ for an integer a.

Proof: According to L'Hopital's rule,

$$\lim_{n \to \infty} \frac{n^a}{2^n} = 0$$

Thus, $n^a < 2^n$ for large enough n.

Big-O Estimates for Some Functions

Prove $n^a = O(2^n)$ for an integer a.

Proof: According to L'Hopital's rule,

$$\lim_{n \to \infty} \frac{n^a}{2^n} = 0$$

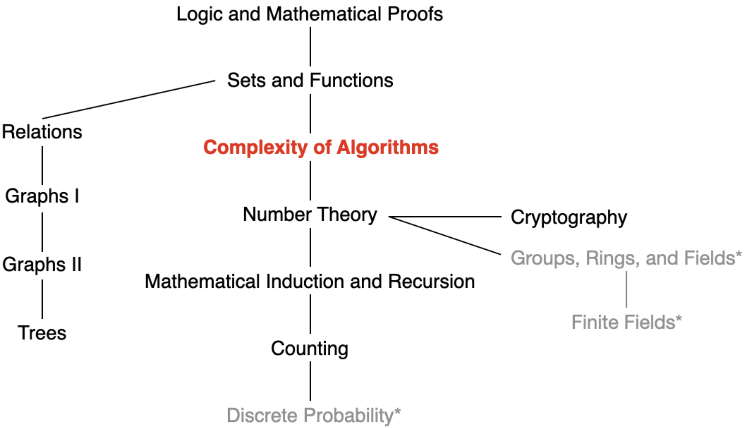
Thus, $n^a < 2^n$ for large enough n.

Note: If f and g are positive-valued functions such that

$$\lim_{n \to \infty} \frac{f(x)}{g(x)} = C < \infty,$$

then $f(x) < (C+1)g(x)$ for large enough x. So $f(n) = O(g(n))$. If that limit is ∞, then $f(n)$ is not $O(g(n))$.

This Lecture



The growth of functions, complexity of algorithm, P and NP problem, ...