Math Review with Solutions

Algorithms CS435

This document provides review material in mathematics to prepare students for a course on Algorithms. This material contains a subset of the topics found in Discrete Mathematics and Calculus I courses (note that both of these courses are pre-requisites for Compro students).

Here is an outline of topics you should know. This document covers only some of these:

(1) High school algebra, including:

  (a) inequalities (and proving inequalities)

  (b) logarithms and laws of logarithms

  (c) laws of exponents

  (d) inverse relationship between logs and exponential functions

(2) Set algebra including De Morgan's Laws

(3) Counting, including combinations and permutations

(4) Propositional logic, logical connectives, truth tables

(5) Proof by mathematical induction

(6) Basic number theory, including

  (a) divisibility and the Division Algorithm (see below)

  (b) prime numbers and unique factorization

  (c) gcd and Bezout's lemma: For any integers a, b, if $g = \gcd(a, b)$, then there are integers x, y so that $ax + by = d$ if and only if $g \mid d$.

  (d) modular notation and equivalent formulations; in particular, $a \equiv b \pmod{n}$ if and only if $n \mid (a - b)$.

  (e) Fibonacci numbers

(7) Derivatives, including

  (a) derivatives of polynomials

  (b) quotient rule, product rule, chain rule

  (c) derivatives of logs and exponential functions

(d) first derivative test

(8) Limits, including limits at infinity and L'Hopital's rule

1. Laws of Logarithms

$$\begin{array}{rcl}
y = \log_b x & \text{means} & x = b^y \\
\log x & \text{means} & \log_2 x \\
\log^n x & \text{means} & (\log x)^n \\
\ln x & \text{means} & \log_e x \ (e \approx 2.71828) \\
\log_b(xy) & = & \log_b x + \log_b y \\
\log_b(x^y) & = & y \log_b x \\
\log_b(\frac{x}{y}) & = & \log_b x - \log_b y \\
\log_b x & = & \dfrac{\log_a x}{\log_a b} \\
0 < x < y & \Rightarrow & \log_b(x) < \log_b(y) \ (\log_b \text{ is } \textit{increasing}) \\
\log_b 1 & = & 0 \\
\log_b b & = & 1 \\
\log_b x & < & x \ (\text{for } b \geq 2 \text{ and } x > 0) \\
\log 1024 & = & 10 \\
\ln 2 & \approx & .693 \ (\text{in particular, } 0 < \ln 2 < 1) \\
\log e & \approx & 1.44 \ (\text{in particular, } 1 < \log e < 2)
\end{array}$$

Problem L1. Show that the following is not true in general, for $k > 1$:

$$(\log n)^k = k \log n.$$

Problem L2. Show that the following is not true in general:

$$\log_b(x + y) = \log_b x + \log_b y$$

Problem L3. Show that, for all $n > 2$,

$$n < n \log n < n^2.$$

It is also true that for all $n > 4$, $n^2 < 2^n$. This is proved by induction. See Problem MI2.

Problem L4. Solve for n:

$$2^{3n-1} = 32.$$

Problem L5. Try this one if you have had a course in calculus. Show that

$$\lim_{n \to \infty} \frac{\log n}{\ln n}$$

is a number between 1 and 2.

2. Sets

(A) A *set* is a collection of objects (a more precise formulation is possible but not necessary for this course).

(∘) The notation $x \in A$ signifies that x is an element of A.

(∘) *Set notation.* The set containing just the elements 1, 2, 3 is denoted $\{1, 2, 3\}$. Elliptical notation can be used to denote larger sets, such as $\mathbf{N} = \{1, 2, 3, \ldots\}$. Set-builder notation defines a set by specifying properties; for instance:

$$E = \{n \mid n \text{ is a natural number and for some } x,\ n = 2 * x\}.$$

(∘) Two sets are *equal* if and only if they have the same elements. Therefore, duplicate elements are not allowed in a set when viewed as a data structure.

(B) B is a *subset* of A, $B \subseteq A$, if every element of B is also an element of A. The empty set, denoted $\emptyset$, is a subset of every set (but is *not* an element of every set!).

(C) If A and B are sets, $A \cup B$ ("the union of A and B") consists of all objects that belong to at least one of A and B; and $A \cap B$ ("the intersection of A and B") consist of all objects that belong to both A and B. Example:

$$\begin{array}{rcl} \{1,2,3\} \cup \{2,3,4\} & = & \{1,2,3,4\} \\ \{1,2,3\} \cap \{2,3,4\} & = & \{2,3\} \end{array}$$

(D) Suppose each of A, B is a set. Then A, B are *disjoint* if A and B have no element in common (that is, $A \cap B = \emptyset$). Similarly, $A_i (i \in I)$ are disjoint if no two of the sets have an element in common.

(E) The *cardinality* or *size* of a set A is denoted $|A|$. Example: $|\{2, 7, 14\}| = 3$.

(F) The *power set* of a set A, denoted $\mathcal{P}(A)$, is the set whose elements are all the subsets of A. Note: If A has n elements, $\mathcal{P}(A)$ has 2^n elements. That is, a set with n elements has 2^n subsets.

(G) If a set A having n elements is totally ordered, then a *permutation* of A is a re-arrangement of the elements of A.

(∘) Example: The following are two of the permuations of $\{1, 2, 3, 4\}$:

$$[1, 2, 4, 3], [4, 3, 2, 1]$$

(∘) The permutation of A that does not re-arrange any of the elements is called the *identity permutation.*

(∘) The number of permutations of an n-element set is $n! = n \cdot (n-1) \cdot (n-2) \cdots 2 \cdot 1$.

(H) The notation $C(n, m)$ is read "the number of combinations of n things taken m at a time" and can be understood to mean "the number of m-element subsets of an n-element set."

(∘) For small values of n, m, $C(n, m)$ can be computed by inspection. Example: Compute $C(3, 2)$. To do the computation, take any 3-element set $\{a, b, c\}$ and write out the 2-element subsets:

$$\{\{a, b\}, \{b, c\}, \{a, c\}\}.$$

The resulting collection now contains 3 two-element subsets of $\{a, b, c\}$. Therefore, $C(3, 2) = 3$.

(∘) Formula for computing $C(n, m)$

$$C(n, m) = \frac{n!}{m!(n-m)!}.$$

Example:

$$C(10, 2) = \frac{10!}{2!8!} = \frac{10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdots 2 \cdot 1}{(2 \cdot 1)(8 \cdot 7 \cdot 6 \cdots 2 \cdot 1)} = \frac{10 \cdot 9}{2 \cdot 1} = 45.$$

(I.) The notation $P_{n,m}$ is read "the number of permutations of n things taken m at a time." The meaning is this: We have a set S with n elements, and we want to arrange m of the elements of S in a particular order. The number of ways to do this is $P_{n,m}$.

(∘) The computation is easier to understand in a simple case. We want to compute $P_{3,2}$. Let $S = \{a, b, c\}$. We want to arrange two elements from S in a particular order. We can think that there are two "slots" to fill—positions 1 and positions 2—with elements from S:

$$\overline{\quad 1 \quad} \qquad \overline{\quad 2 \quad}$$

To fill these slots, we perform two tasks in succession:

Task 1: Pick a 2-element subset from S
Task 2: Arrange it so one element is in position 1, the other in position 2.

There are $C(3, 2)$ ways to perform Task 1. After a set has been selected, there are 2! ways to arrange that set—that is, 2! ways to place the elements into position 1 and position 2. Therefore:

$$P_{3,2} = C(3, 2) \cdot 2!$$

(∘) The same logic gives the formula for $P_{n,m}$:

$$P_{n,m} = C(n, m)m! = \frac{n!}{(n-m)!}.$$

(∘) Example: Compute $P_{10,2}$.

$$P_{10,2} = \frac{10!}{(10-2)!} = \frac{10!}{8!} = 10 \cdot 9 = 90.$$

Problem S1. Are the following sets equal? Explain.

$$\{1, 1, 2\}, \{1, 2\}, \{2, 1\}.$$

Problem S2. Is the following statement true or false?

$$\{1, \{2, 3\}\} \subseteq \{1, 2, 3, 4, 5, \ldots\}$$

Problem S3. What is the powerset of the set $\{1, 2\}$?

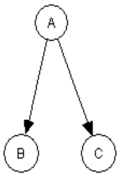
Problem S4. List all the permutations of the set $\{1, 3, 4\}$.

Problem S5. In how many ways can 5 students, from a group of 9 students, be seated in a row of 5 chairs?

Problem S6. A committee of three representatives is to be chosen from a larger group of 20 people. In how many ways can this committee be formed?

3. Directed Graphs and Functions

A directed graph is a set of objects (called *vertices* or *nodes*) together with a set of arrows that join some of the vertices. Here is a simple example:



A function from a set X to a set Y—written $f : X \to Y$—is a special kind of directed graph f (we usually denote functions using typical letters f, g, h, etc.) with the following characteristics:

- (◦) The objects of the graph f are the elements of X together with the objects of Y.
- (◦) Each arrow of f always starts at an element of X and points to an element of Y. If, in f, x points to y, we write $x \to y$ or $f(x) = y$.
- (◦) In f, no $x \in X$ ever points to more than one element of Y
- (◦) In f, every element of X does point to *at least one* element of Y.

When $f : X \to Y$ is a function, X is called its *domain*, Y its *codomain*.

Concepts Related to Functions. Suppose $f : X \to Y$ is a function.

1. *Onto*. A f is *onto* if for each $y \in Y$ there is an element $x \in X$ so that $x \to y$.
2. *Range*. The range of f is the set of all $y \in Y$ that are pointed to by one or more x in X; the range is the set of all *output values* of f. If the range of f is Y itself, f is onto.

(3) *1-1.* A function $f : X \to Y$ is *1-1* if, whenever x and x' are distinct elements of X, and $x \to y$ and $x' \to y'$, then y and y' are also distinct elements of Y.

Example. Let $A = \{1, 2, 3\}$ and $B = \{4, 5, 6\}$. Define $f : A \to B$ as follows:

A	f	B
1	→	4
2	→	5
3	→	6

In other words, in the directed graph f, $1 \to 4, 2 \to 5, 3 \to 6$. Another way to say this is that f takes the number 1 to 4, the number 2 to 5, and the number 3 to 6. Here is notation that can be used to state this fact:

$$\begin{aligned} f(1) &= 4 \\ f(2) &= 5 \\ f(3) &= 6 \end{aligned}$$

Example. We define a function g, also having domain A and codomain B (defined in the previous example), as follows:

$$\begin{aligned} g(1) &= 4 \\ g(2) &= 4 \\ g(3) &= 4 \end{aligned}$$

Here g is also a function. In the previous example, the function f was *1-1*—no two elements of the domain were assigned the same value by f. Clearly, g does not have that property; in fact, all elements of the domain A of g are assigned the single value 4.

Example. Returning to the functions f and g of the previous examples, notice that the range of f is precisely equal to B, so f is onto. On the other hand, the range of g is just the singleton set $\{4\}$, and so g is *not* onto.

Problem DGF1. Consider the function $f(n) = n^2$, where the domain of f is the set $\mathbf{N}$ of all natural numbers. Is f 1-1? What is the range of f? Is f onto?

Some functions from $\mathbf{N}$ to $\mathbf{N}$ have the convenient property of being *increasing*. This means that as input values increase, output values also increase. More precisely, we have the following definition:

Definition. A function $f : \mathbf{N} \to \mathbf{N}$ is said to be *increasing* if, whenever $m < n$, we have $f(m) < f(n)$. A function $g : \mathbf{N} \to \mathbf{N}$ is said to be *nondecreasing* if, whenever $m < n$, we have $g(m) \leq g(n)$.

Example. Obviously, the identity function $f(n) = n$ is increasing. It is equally easy to see that the function $g(n) = kn$ for any integer $k > 1$ is also increasing. This can be verified by simple algebra: if $m < n$, then multiplying on both sides by k gives us $km < kn$, which establishes that $g(m) < g(n)$.

Problem DGF2. Show that the function $f(n) = n^2$, with domain $\mathbf{N}$, is increasing.

4. Summations

$$\begin{aligned}
\sum_{i=1}^{N} 1 &= N \\
\sum_{i=1}^{N} i &= \frac{N(N+1)}{2} \\
\sum_{i=1}^{N} i^2 &= \frac{N(N+1)(2N+1)}{6} \\
\sum_{i=0}^{N} 2^i &= 2^{N+1} - 1 \\
\sum_{i=0}^{N} a^i &= \frac{a^{N+1} - 1}{a - 1} \\
\sum_{i=0}^{N} a^i &< \frac{1}{1-a} \text{ (whenever } 0 < a < 1) \\
\sum_{i=1}^{N} \frac{1}{i} &\approx \ln 2 \, \log N \text{ (the difference between these falls below 0.58 as } N \text{ tends to infinity)}
\end{aligned}$$

Problem SUM1. Rewrite the following in terms of the variable N, using the formulas above.

$$\sum_{i=1}^{N} 2i^2 + 3i - 4.$$

5. Mathematical Induction

Mathematical induction is a technique for proving mathematical results having the general form "for all natural numbers n, . . ." For example, suppose you would like to prove that for all natural numbers $n > 1$, $n^2 > n+1$. You might try a few values for n to see if the statement makes sense. Certainly $2^2 > 2+1, 3^2 > 3+1, 10^2 > 10+1$. These examples suggest that the statement always holds true. But how do we know for sure? It is at least conceivable that for certain very large numbers that we are unlikely to consider, the statement is no longer true. Mathematical induction is a technique for demonstrating that such a formula must hold true for every natural number > 1, without exception.

The intuitive idea behind Mathematical Induction is this: Suppose you wish to prove that some statement $\phi(n)$, which asserts something about each whole number n, is true for every n. For example, to prove that for all $n \geq 0$, $n < 2^n$, we would use "$n < 2^n$" as our statement $\phi(n)$. We wish to show that this statement holds for every n. Suppose now that we can prove two things:

(1) that $\phi(0)$ is true (in our example, this would mean that we can prove $0 < 2^0$);

(2) that, for any n, if $\phi(n)$ happens to be true, then $\phi(n+1)$ must also be true (in our example, this would mean that, if it happens to be true that $n < 2^n$, then it must be true that $n+1 < 2^{n+1}$).

Mathematical Induction says that, if you can prove both (1) and (2), then you have proven that, for every n, $\phi(n)$ is indeed true.

Below are several forms of induction. Each provides a valid approach to proving the correctness of a statement about natural numbers. Different forms are useful in different contexts. We include an example of each.

Standard Induction. Suppose $\phi(n)$ is a statement depending on n. If

(•) $\phi(0)$ is true, and

(•) under the assumption that $n \geq 0$ and $\phi(n)$ is true, you can prove that $\phi(n+1)$ is also true,

then $\phi(n)$ holds true for all natural numbers n.

In Standard Induction, the step in the proof where $\phi(0)$ is verified is called the *Basis Step*. The second step, where $\phi(n+1)$ is proved assuming $\phi(n)$, is called the *Induction Step*. As we reason during this second step, we will typically need to make use of $\phi(n)$ as an assumption; in this context, $\phi(n)$ is called the *induction hypothesis*.

Note. Standard Induction allows you to establish that a statement $\phi(n)$ holds for all natural numbers 0,1,2, ... However, sometimes the objective is to show that $\phi(n)$ holds for all numbers n that are larger than a fixed number k. Standard Induction may still be used. Here is a precise statement:

Standard Induction (General Form). Let $k \geq 0$. Suppose $\phi(n)$ is a statement depending on n. If

(•) $\phi(k)$ is true, and

(•) under the assumption that $n \geq k$ and $\phi(n)$ is true, you can prove that $\phi(n+1)$ is also true,

then $\phi(n)$ holds true for all natural numbers $n \geq k$.

Problem MI1. Prove that, for every natural number $n \geq 1$,

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

Problem MI2. Show that for every natural number $n > 4$, $n^2 < 2^n$.

Total Induction. Suppose $\phi(n)$ is a statement depending on n and $k \geq 0$. If

(•) $\phi(k)$ is true, and

(•) under the assumption that $n > k$ and that each of $\phi(k), \phi(k+1), \ldots, \phi(n-1)$ are true, you can prove that $\phi(n)$ is also true,

then $\phi(n)$ holds true for all $n \geq k$.

Problem MI3. Prove that if $f(n) = 2^n$, then f is increasing.

Finite Induction. Suppose $0 \leq k \leq n$, and suppose $\phi(i)$ is a statement depending on i, where $k \leq i \leq n$. If

(•) $\phi(k)$ is true, and

(•) under the assumption that $k \leq i < n$ and that $\phi(i)$ is true, you can prove $\phi(i+1)$ is true,

then $\phi(i)$ holds true for all i with $k \leq i \leq n$.

Note. Another equally valid variant of Finite Induction uses an induction hypothesis that is essentially the same as the one used for Total Induction.

Problem MI4. The following is a Java method for computing $n!$ for any n.

```
int factorial(int n) {
  if(n==0 || n==1) return 1;
  int accum = 1;
  for(int i = 2; i <= n; ++i) {
    accum *= i;
  }
  return accum;
}
```

Prove that for every n, the output of `factorial(n)` is $n!$.

6. Basic Number Theory

We review some basics about number theory. Assume $a, b, c, \ldots$ are integers.

- (●) [*divides*] $a \mid b$ means a divides b, i.e., for some c, $b = ac$
- (●) [*floor and ceiling*] $\lfloor a \rfloor$ is the largest integer not greater than a ($\lfloor\cdot\rfloor$ is called the *floor function*) and $\lceil a \rceil$ is the smallest integer not less than a ($\lceil\cdot\rceil$ is called the *ceiling function*).

Note. The floor function applied to rational numbers a/b yields the same results as Java's integer division when both a and b are positive. However, when one is negative and the other positive, the results differ:

$$\begin{array}{rcll} -5/4 & = & -(5/4) = -1 & \text{(Java integer division)} \\ \lfloor -5/4 \rfloor & = & -2 & \text{(mathematics)} \end{array}$$

- (●) [*greatest common divisor*] $c = \gcd(a, b)$ means c is the largest integer that divides both a and b
- (●) [*least common multiple*] $c = \text{lcm}(a, b)$ means c is the smallest integer for which $a \mid c$ and $b \mid c$
- (●) [*modulus*] If $a > 0$, then $b \bmod a$ equals the (nonnegative) remainder on dividing b by a. ($b \bmod a$ is a nonnegative number less than a.)

  Note. Java's mod function `%` is the same as mod for positive inputs, but if $a, b > 0$, then $-a \,\%\, b = -(a \bmod b)$.

  - (○) Example: $8 \% 3 = 8 \bmod 3 = 2$
  - (○) Example: $-8 \bmod 3 = 1$ but $-8 \% 3 = -(8 \bmod 3) = -2$

- (●) [*congruence*] $b \equiv a (\bmod n)$ means $b \bmod n = a \bmod n$. Equivalently $n \mid (b - a)$ (see one of the examples below for a proof of this equivalence).

- (●) **The Division Algorithm.** For each pair of integers a, b with $a > 0$, there is a unique pair q, r such that

  - (○) $b = aq + r$ (q is the *quotient*, r is the *remainder*), and
  - (○) $0 \le r < a$.

  Moreover, $q = \lfloor \frac{b}{a} \rfloor$ and $r = b \bmod a$.

  Note. The equation $b = aq + r$ also holds with $q = \frac{b}{a}$ (integer division) and $r = b \,\%\, a$, but in the case where $a > 0$ and $b < 0$, it turns out that $r < 0$ (so the inequality $0 \le r < a$ given above fails if these computations are used).

- (●) [primes] A positive integer p is *prime* if its only positive divisors are 1 and p. A positive integer c is *composite* if there are positive integers m, n, both greater than 1, such that $c = m \cdot n$.

  Example. Show that every integer > 1 is a product of primes. (A prime itself is considered a product of primes.)

Solution. Proceed by induction on natural numbers $n \geq 2$. Since 2 is prime, 2 is a product of primes. This takes care of the base case. Proceeding with Total Induction, assume $n > 2$ and every number $< n$ is a product of primes. Consider n. If n is already prime, we are done. If n is composite, $n = m \cdot k$, then since both m, k are $< n$, by the induction hypothesis, each of m, k is a product of primes. It follows that n is a product of primes. This completes the induction and the proof.

Example. Prove that there are infinitely many primes.

Solution. Suppose there were only finitely many primes. Let $p_0, p_1, p_2, \ldots, p_m$ be a list of all primes in increasing order. Let P be the product of these primes; that is, let $P = p_0 \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_m$. Since $P + 1$ is larger than all the primes in the list, $P + 1$ must be composite and we can write $P + 1 = k \cdot n$ for some k, n. By the previous example, k must be a product of primes; in particular, some p_i divides k; it follows that p_i divides $P+1$. But p_i also divides P (recall the definition of P). Hence, p_i divides the difference $(P+1)-P$, which is impossible. Therefore, there cannot be only finitely many primes.

(•) *Fibonacci Numbers.* The sequence $F_0, F_1, F_2, \ldots, F_n, \ldots$ of Fibonacci numbers is defined by

$$\begin{aligned} F_0 &= 0; \\ F_1 &= 1; \\ F_n &= F_{n-1} + F_{n-2}. \end{aligned}$$

Problem BNT1. Let a, b be integers, not both 0.

(A) Suppose $g = \gcd(a, b)$. Show that there are integers x, y so that $g = ax + by$. *Hint.* Let $S = \{z \mid z \text{ is positive, having the form } ax + by \text{ for some } x, y\}$, and let $d = \min S$. First show that $d|a$ and $d|b$. Then show that $d = \gcd(a, b)$ by showing that $d \geq c$ for any common divisor c of a, b.

(B) Suppose there are integers x, y, c such that $c > 0$ and $ax+by = c$. Show that if $g = \gcd(a, b)$ then $g|c$. Hint: Use the Hint for part (A) and make use of the Division Algorithm.

Problem BNT2. Suppose $g = \gcd(a, b)$ and suppose d is some other common divisor of a, b; that is, suppose $d \mid a$ and $d \mid b$. Show that $d \mid g$.

Problem BNT3. Show that $a \equiv b \bmod n$ if and only if $n \mid (a - b)$. *Hint.* Try writing

$$\begin{aligned} a &= \lfloor \frac{a}{n} \rfloor \cdot n + a \bmod n \\ b &= \lfloor \frac{b}{n} \rfloor \cdot n + b \bmod n \end{aligned}$$

Subtracting, you get

$$(*) \qquad a - b = \big(\lfloor \frac{a}{n} \rfloor \cdot n - \lfloor \frac{b}{n} \rfloor \cdot n\big) + \big(a \bmod n - b \bmod n\big).$$

This observation will help in the proof in both directions.

Problem BNT4. Find the unique q and r guaranteed by the Division Algorithm, where $a = 7$ and $b = -20$; that is, find q, r with $b = aq + r$ and $0 \leq r < a$. Then obtain values q' and r' such that $b = aq' + r'$ that makes use of Java's mod function.

Problem BNT5—Extra Credit. Prove the following: For all nonzero a, b, $\gcd(a, b) \cdot \mathrm{lcm}(a, b) = |ab|$. If you want to try this one, proceed by proving Lemmas 1 and 2 below:

Lemma 1. Show that if a, b are nonzero integers and $\ell = \mathrm{lcm}(a, b)$, then the rational number $\frac{ab}{\ell}$ is an integer.

Lemma 2. Show that if a, b are nonzero integers and $\ell = \mathrm{lcm}(a, b)$, then the integer $\frac{ab}{\ell}$ divides both a and b.

7. Propositional Logic.

Propositional logic provides the tools for answering the following question: Suppose you have two statements p, q in mind. If we know the truth values of each of p and q—that is, we know whether each is true or false—what can we say about the truth values of new statements built up from these using logical connectives 'and,' 'or,' and 'not'? To begin, we consider truth values of p *and* q and the statement p *or* q.

For concreteness, let's use the following statements for p and q:

$$\begin{aligned} p &= \text{'The dining room table has some boxes on it.'} \\ q &= \text{'One of the tires on my car is flat.'} \end{aligned}$$

Each of p and q is a legitimate statement in the sense that it makes sense to ask whether either is true or false. By the way, some examples of expressions in English that are *not* legitimate statements in this sense would be 'Close the door!' and 'Why don't I have more money?' since neither expression can be said to have a truth value—a value of 'true' or 'false.' Notice that 'p and q' is the new statement

'The dining room table has some boxes on it and one of the tires on my car is flat,

and 'p or q' is another new statement

'The dining room table has some boxes on it or one of the tires on my car is flat.'

If both p and q happen to be true, notice that the new statement 'p and q' is also true. However, if even one of the statements p, q is false, this fact will force the new statement 'p and q' to be false too.

"What about the statement 'p or q'? As we have seen informally in some of our proofs so far, in logic, the connective 'or' has a specialized meaning, roughly equivalent to the term 'and/or' in English. To say 'p or q' is true means—as it is used in logic—that *either p is true or q is true or both are true*

In English, the word 'or' often has the meaning 'either p or q is true, but not both'—we will not use this meaning of 'or.'

With this understanding of 'or,' if both p and q happen to be true, the new statement 'p or q' is also true. In fact, if one or both of p, q happen to be true, 'p or q' must also be true. The only way 'p or q' could be false is if both p *and* q are false.

Our analysis, which applies to any possible statements p and q, leads to the following tabluation that gives a complete description of the truth values that result from applying the connectives 'and' and 'or.' In logic, the connective 'and' is denoted by the symbol '$\wedge$' while 'or' is denoted by '$\vee$.'

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Using these tables, we can create new tables that describe the truth values of more complicated statements, formed by combining multiple statements $p, q, r, \ldots$ using the connectives $\wedge$ and $\vee$. For instance, given statements p and q, here is a truth table for the new statement $(p \wedge q) \vee p$:

p	q	$p \wedge q$	$(p \wedge q) \vee p$
T	T	T	T
T	F	F	T
F	T	F	F
F	F	F	F

This table is the same as the table for $p \wedge q$, except that we have added one new column, which lists the truth values for $(p \wedge q) \vee p$. Since this statement is obtained by combining $(p \wedge q)$ with p (using $\vee$), we fill in this final column by looking down the two columns, headed '$(p \wedge q)$' and 'p,' and computing the appropriate truth values, making use of the rules given in the truth table for the 'or' ($\vee$) connective. For instance, in the first row, we see a 'T' in both the '$(p \wedge q)$'

column and the 'p' column; by the rules for 'or,' 'true or true' gives us 'true,' and we place a 'T' in the final column for that row. In the second row, there are 'F' and 'T,' and this combination gives us a 'T' again. The computation continues in this way. The final column of the new truth table tells us the possible truth values of the statement $(p \wedge q) \vee p$ for any combination of truth values of p and q individually.

Now notice that in this new truth table, the truth values that appear in the final column are identical, row for row, with the truth values that appear in the 'p' column. This fact tells us that the statement p is true exactly when $(p \wedge q) \vee p$ is true, and false exactly when $(p \wedge q) \vee p$ is false. In other words, the two statements are *equivalent*. This observation is extremely useful; it gives us a way of determining when two statements, built from the same basic propositions $p, q, r, \ldots$, are equivalent. We summarize this point below by giving an official definition of equivalent propositions, and then recording our observation as a theorem.

Definition 1 (Equivalent Propositions)**.** *Suppose two statements S_1 and S_2 are built up by combining the same basic propositions $p, q, r, \ldots$ with connectives $\vee, \wedge, \ldots$. Then we say S_1 and S_2 are* **equivalent** *if S_1 is true if and only if S_2 is true.*

Theorem 1 (*Equivalent Propositions Theorem*)**.** *Suppose two statements S_1 and S_2 are built up by combining the same basic propositions $p, q, r, \ldots$ with connectives $\vee, \wedge, \ldots$. Then S_1 and S_2 are equivalent if in a truth table for both S_1 and S_2, they have, row for row, identical columns.*

An important point to remember as we examine this last truth table once again is that the facts that it reveals hold true regardless of the propositions p and q that we start with. In particular, for any choice of p and q, the statements $(p \wedge q) \vee p$ and p are always equivalent.

We can apply our observation about this logical equivalence to a problem about sets: Consider the following problem: Show that for any sets A, B,

$$(A \cap B) \cup A = A.$$

We can rewrite this set equation in a corresponding logical form by observing that the sets on the left and right are equal if and only if for any x,

$$((x \in A \text{ and } x \in B) \text{ or } x \in A) \text{ if and only if } x \in A.$$

This logical form suggests a clever application of the truth table given above. Let p be the statement '$x \in A$' and let q be the statement '$x \in B$.' We can then make the following translation:

$$\begin{array}{rcl} p & \text{becomes} & \text{“}x \in A\text{”} \\ (p \wedge q) \vee p & \text{becomes} & \text{“}(x \in A \wedge x \in B) \vee x \in A.\text{”} \end{array}$$

But from our truth table analysis, we know that the statements $(p \wedge q) \vee p$ and p are equivalent. Therefore,

$$(x \in A \wedge x \in B) \vee x \in A \text{ is true if and only if } x \in A \text{ is true.}$$

But this is simply the logical form of the assertion that $(A \cap B) \cup A = A$.

Therefore, we have applied the fact that $(p \wedge q) \vee p$ is equivalent to p to show a fact about sets—that for any A, B, $(A \cap B) \cup A = A$.

We would like to use our new skills in logic to show next that, for any sets A, B, C, we have

$$(A \cap B) \cup C = (A \cup C) \cap (B \cup C).$$

This equality can be expressed in the following logical form:"

$$((x \in A \text{ and } x \in B) \text{ or } x \in C)$$
$$\text{if and only if}$$
$$(x \in A \text{ or } x \in C) \text{ and } (x \in B \text{ or } x \in C).$$

This form suggests the use of *three* basic propositions this time."

$$\begin{aligned} p &= \text{“}x \in A\text{”} \\ q &= \text{“}x \in B\text{”} \\ r &= \text{“}x \in C.\text{”} \end{aligned}$$

Translating the logical form in terms of p, q, r, our logical problem becomes the following."

Prove that the statement $(p \wedge q) \vee r$ *is equivalent to the statement* $(p \vee r) \wedge (q \vee r)$.

To accomplish this task, we must build a truth table for these two statements, using the basic propositions p, q, r.

The first step in building the new truth table is to fill the columns headed by p, q, r with Ts and Fs. The objective is to list every possible combination of truth values for these three propositions. For instance, we could have p true, q false, and r true; we could have p false, q false, and r true; and we could have many other possible combinations. In fact, since there are just two choices for each of the propositions (either T or F) and there are three propositions (p, q, r), there are exactly $2 \cdot 2 \cdot 2 = 8$ possible combinations of Ts and Fs. We list the possible combinations below.

p	q	r
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F

There is a convenient scheme for setting up these three columns. Since there must be 8 rows, each column must have 4 Ts and 4 Fs. For the left column, begin by listing all 4 Ts, then list all

4 Fs. In the middle column, alternate pairs of Ts (TT) and Fs (FF). And in the third column, alternate Ts and Fs. A similar scheme can be used to get started with 4, 5, or any number of basic propositions.

We now build a truth table that will include the two statements we wish to prove equivalent. We will need to provide columns for each of the 'subclauses' of these statements—namely, $p \wedge q$, $p \vee r$, and $q \vee r$—so we can gradually build up the final statements and compute their truth values. Here is the truth table with all the subclauses accounted for.

p	q	r	$p \wedge q$	$p \vee r$	$q \vee r$
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	F	T	T
T	F	F	F	T	F
F	T	T	F	T	T
F	T	F	F	F	T
F	F	T	F	T	T
F	F	F	F	F	F

Notice how the last three columns were obtained. The $p \wedge q$ column was obtained by comparing the truth values, row by row, of the columns headed by p and q, using the '$\wedge$' connective. Thus, in the fourth row where p is true and q is false, we place an F in the $p \wedge q$ column. Similarly, the $p \vee r$ column is obtained by comparing, row by row, the columns headed by p and r, this time using the '$\vee$' connective. A similar observation applies to the final column, headed by $q \vee r$.

Finally, we use these new columns to compute the truth values of our final statements. Here is our final truth table:

p	q	r	$p \wedge q$	$p \vee r$	$q \vee r$	$(p \wedge q) \vee r$	$(p \vee r) \wedge (q \vee r)$
T	T	T	T	T	T	T	T
T	T	F	T	T	T	T	T
T	F	T	F	T	T	T	T
T	F	F	F	T	F	F	F
F	T	T	F	T	T	T	T
F	T	F	F	F	T	F	F
F	F	T	F	T	T	T	T
F	F	F	F	F	F	F	F

The column headed by $(p \wedge q) \vee r$ is obtained by comparing the $p \wedge q$ column with the r column (using $\vee$), and the column headed by $(p \vee r) \wedge (q \vee r)$ is obtained by comparing the $p \vee r$

column with the $q \vee r$ column (using $\wedge$).

Now notice that the final two columns are identical. This shows that our two statements are equivalent. Since to say that the logical form of the equality is true is the same as saying that the equality itself is true, we may conclude that the sets $(A \cap B) \cup C$ and $(A \cup C) \cap (B \cup C)$ are equal for any sets A, B, and C.

Another very simple logical connective is *negation*, represented by the symbol $\neg$. A proposition $\neg p$ is true if and only if p is false. Here is a truth table for $\neg$:

p	$\neg p$
T	F
F	T

Propositional Logic Continued: The Conditional Connective

The conditional connective is indicated by the symbol $\rightarrow$. If p and q are propositions, $p \rightarrow q$ is another proposition, read 'if p then q.' Such a statement is called a conditional statement because it asserts that q is true *under the condition that* p is true. In the statement $p \rightarrow q$, p is called the *antecedent* or *hyothesis* and q is called the *consequent* or *conclusion*. For future reference, he statement $q \rightarrow p$ is called the *converse* of $p \rightarrow q$.

In order for a conditional statement to be *false*, it is necessary for the condition, or antecedent, to be true, but the consequent to be false. This requirement is familiar in English: If I say that 'If I go to the store, I will return home with a carton of milk,' I cannot be said to have broken my word if I never make it to the store, whether or not I come home with milk. The promise can be considered broken only if I do go to the store and still return home without milk. This way of understanding conditional statements is reflected in the following truth table definition of the conditional connective.

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Main Fact About Conditionals. *The only way a conditional statement $p \rightarrow q$ can fail to be true is if p is true and q is false.*

The combination "p true and q false" reflects the one and only way in which the 'promise' implied by the statement $p \rightarrow q$ could be 'broken.'

The Main Fact About Conditionals has several important consequences. One is that, whenever the antecedent is *false*, the conditional statement itself is *true*. Thus, for example, the following statement is indisputably true:

$$\text{If } 2+2=5, \text{ then all dogs are blue,}$$

not because the conclusion is true, but because the hypothesis is false. This may seem strange, but it follows from the definition of the conditional connective. Whenever we come to the conclusion that a conditional statement is true for the trivial reason that the antecedent is known to be false, we say that the conditional is *vacuously true*.

Another consequence of the Main Fact About Conditionals is that it suggests an alternative way to make a conditional statement without using the connective →: Since the negation of $p \to q$ asserts p is true and q is false, we might conjecture that $\neg(p \to q)$ is equivalent to $p \wedge \neg q$. Applying de Morgan's Laws, we can restate our conjecture by saying that $p \to q$ is equivalent to $\neg p \vee q$. The conjecture is correct, and we can prove it using the following truth table.

p	q	$\neg p$	$p \to q$	$\neg p \vee q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

If a conditional statement $S_1 \to S_2$ is a tautology, we say that S_1 *implies* S_2, and we use a special arrow (⇒) to indicate this relationship symbolically. A good example of this relationship can be seen if we extend the above truth table so we can examine the truth values of $(p \to q) \to (\neg p \vee q)$, and also the truth values of its converse, $(\neg p \vee q) \to (p \to q)$.

p	q	$\neg p$	$p \to q$	$\neg p \vee q$	$(p \to q) \to (\neg p \vee q)$	$(\neg p \vee q) \to (p \to q)$
T	T	F	T	T	T	T
T	F	F	F	F	T	T
F	T	T	T	T	T	T
F	F	T	T	T	T	T

The extended truth table establishes the fact that $(p \to q) \Rightarrow (\neg p \vee q)$ and also that $(\neg p \vee q) \Rightarrow (p \to q)$.

The truth table also illustrates the following important connection between equivalent statements: Two statements S_1 and S_2 are equivalent if and only if both $S_1 \Rightarrow S_2$ and $S_2 \Rightarrow S_1$. The latter observation suggests one other logical connective: the *biconditional*, denoted ↔. Given propositions p and q, the statement $p \leftrightarrow q$ is true just in case p and q are equivalent, that is, just in case p and q have identical truth values. The following table for ↔ provides a definition for this connective.

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

If we extend the truth table displayed above that we used for comparing $(p \rightarrow q)$ and $(\neg p \vee q)$ by adding a column for the statement $(p \rightarrow q) \leftrightarrow (\neg p \vee q)$, we see that the biconditional statement is in fact a tautology.

p	q	$\neg p$	$p \rightarrow q$	$\neg p \vee q$	$(p \rightarrow q) \leftrightarrow (\neg p \vee q)$
T	T	F	T	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

This example illustrates the fact that two statements are equivalent if and only if their biconditional is a tautology. We have the following theorem.

Theorem 2 (*Equivalent Statements***).** *Suppose S_1 and S_2 are statements built up from basic propositions using $\wedge, \vee, \neg \rightarrow$, and $\leftrightarrow$. The following statements are equivalent:*

(1) S_1 and S_2 are equivalent; that is, $S_1 \Leftrightarrow S_2$

(2) the statements $S_1 \rightarrow S_2$ and $S_2 \rightarrow S_1$ are tautologies

(3) S_1 and S_2 imply each other; that is, $S_1 \Rightarrow S_2$ and $S_2 \Rightarrow S_1$

(4) the statement $S_1 \leftrightarrow S_2$ is a tautology.

Problem PL-1. Use truth tables to establish the following set identity:

$$(X \cup Y) \cap Z = (X \cap Z) \cup (Y \cap Z).$$

Problem PL-2. **SAT** is an important problem that is considered in complexity theory: A *boolean expression* B is an expression consisting of one or more propositional symbols $p, q, \ldots$ joined by the connectives $\wedge, \vee, \neg$. For instance, $B = B(p, q, r) = p \wedge q \vee (\neg r \wedge \neg p)$ is a boolean expression. **SAT** is the following problem: Given a boolean expression B, is there an assignment of truth values (T or F) to the propositional symbols so that B evaluates to true? Solve the following two instances of **SAT**:

(1) $(p \vee q) \wedge (\neg q \wedge (r \vee p))$

(2) $(p \wedge \neg q) \vee \neg (p \vee (\neg r \wedge q) \vee r)$

Appendix: More On Mathematical Induction

In this Appendix, we will clear up common confusions about induction by building up the idea of induction from a more basic context. For this purpose, we will try to prove, as an illustrative but basic example, that for any $n \in \{1, 2, 3, 4\}$, we have $2^n > n$. One way to do it is to prove each of the following four statements directly.

$$\begin{array}{rcl} 2^1 & > & 1 \\ 2^2 & > & 2 \\ 2^3 & > & 3 \\ 2^4 & > & 4 \end{array}$$

We could reduce the number of statements that we need to prove to just two if we do it like this: First, we prove $2^1 > 1$. Then, we prove the statement:

$$\text{for any } n \in \{1, 2, 3\}, \text{ if } 2^n > n, \text{ then } 2^{n+1} > n + 1.$$

Now let's check that this new approach would actually establish the result. We can break this second statement into three in the following way.

$$\begin{array}{c} \text{if } 2^1 > 1, \text{ then } 2^2 > 2 \\ \text{if } 2^2 > 2, \text{ then } 2^3 > 3 \\ \text{if } 2^3 > 3, \text{ then } 2^4 > 4 \end{array}$$

In our first step we established that $2^1 > 1$. If we combine this with the first conditional statement that we just put on the board, if $2^1 > 1$, then $2^2 > 2$, we can conclude that $2^2 > 2$.
Therefore, assuming once again that we have proved if $2^2 > 2$ then $2^3 > 3$, we may conclude $2^3 > 3$. And then, for the final step, we can combine $2^3 > 3$ with the conditional if $2^3 > 3$ then $2^4 > 4$ to conclude $2^4 > 4$.

These verifications show that this approach to the proof—beginning with a direct proof of $2^1 > 1$—followed by proof of those three conditional statements does indeed give a proof that, for all $n \in \{1, 2, 3, 4\}, 2^n > n$. What actually has to be proved, using this approach, is, first, that $2^1 > 1$, and then each conditional statement—that if $2^1 > 1$, then $2^2 > 2$, and so forth.

The approach so far is cumbersome. What makes it worthwhile is that we will be able to prove all three conditional statements simultaneously. We will do this by proving the following statement:

$$(*) \qquad \text{for any } n \in \{1, 2, 3\}, \text{ if } 2^n > n, \text{ then } 2^{n+1} > n + 1.$$

We discuss how to create a proof for such a statement. We begin by noticing that, for all $n \in \mathbf{N}$, $2n \geq n + 1$: Since $n \in \mathbf{N}$, we know $n \geq 1$. We can add n to both sides, and the equality still holds. So we get $n + n \geq n + 1$. Since $2n = n + n$, we get $2n \geq n + 1$.

To prove $(*)$, suppose $n \in \{1,2,3\}$ and assume that $2^n > n$. Then

$$(+) \qquad 2^{n+1} = 2 \cdot 2^n > 2 \cdot n \geq n+1$$

The statement $2^{n+1} = 2 \cdot 2^n$ follows by the definition of exponentiation. The statement $2 \cdot 2^n > 2 \cdot n$ is obtained by multiplying both sides of $2^n > n$ (which we are assuming is true) by 2. And the final step, $2 \cdot n \geq n+1$, was what we proved first.

In this argument, we have established three conditionals in one argument. The point of all these efforts is to make it clear that the same thing works when we attempt to prove $2^n > n$ for *all* $n \in \mathbf{N}$. We can first show that $2^1 > 1$. And for the second step, we can show for all $n \in \mathbf{N}$, if $2^n > n$, then $2^{n+1} > n+1$. In this case, establishing that second step establishes *infinitely many* conditional statements in a single argument. And the proof of that second step is exactly the same as the argument we just gave in $(+)$.

From these considerations, we restate the Principle of Mathematical Induction, which tells us that reasoning of this kind produces valid arguments in proofs of statements of the form 'for all $n \in \mathbf{N}, \phi(n)$.'

Principle of Mathematical Induction. Suppose $\phi(n)$ is a formula, where the parameter n stands for a natural number. Suppose the following two statements have been proved:

(1) (*Basis Step*) $\phi(1)$ is true.

(2) (*Induction Step*) For each $n \in \mathbf{N}$, if $\phi(n)$ is true, then $\phi(n+1)$ is also true.

Then, for all $n \in \mathbf{N}$, $\phi(n)$ is true.

The Principle of Mathematical Induction is just a formal expression of the procedure we just went through to prove that for every $n \in \mathbf{N}, 2^n > n$. In the argument we did, the first step, which we now call the *Basis Step*, was to establish $2^1 > 1$. The next step, which is called the *Induction Step*, is to prove, for any $n \in \mathbf{N}$, that, assuming $2^n > n$, then $2^{n+1} > n+1$ as well. As we perform our reasoning during the Induction Step, each time we make use of the assumption that $2^n > n$, we are *using the induction hypothesis* at that point.

Answers to Problems

L-ProblemsProblem L1. To show this, it is enough to give an example for which the equation does not hold true. Consider $n = 4$ and $k = 3$. Then $(\log n)^k = (\log 4)^3 = 2^3 = 8$, but $k \log n = 3 \log 4 = 3 \cdot 2 = 6$.

Problem L2. To show this, it is enough to give an example for which the equation does not hold true. Consider $b = 2, x = 4, y = 4$. Then $\log_b(x + y) = \log(4 + 4) = \log 8 = 3$, but $\log_b x + \log_b y = \log 4 + \log 4 = 2 + 2 = 4$.

Problem L3. We show that, for all $n > 2$,

$$n < n \log n < n^2.$$

Since $2 < n$ and logs preserve order ($x < y \Rightarrow \log x < \log y$), we have $1 = \log 2 < \log n$, so, multplying by n, $n < n \log n$. To prove $n \log n < n^2$, recall lthat for all $n \geq 2$, $\log n < n$, so, multiplying both sides by n gives the result.

Problem L4. We solve for n:

$$2^{3n-1} = 32.$$

Applying log to both sides yields

$$3n - 1 = 5.$$

Solving, we get $n = 2$.

Problem L5. We show that

$$\lim_{n\to\infty} \frac{\log n}{\ln n}$$

is a number between 1 and 2. Using properties of logarithms listed above, we have

$$\begin{aligned}\lim_{n\to\infty} \frac{\log n}{\ln n} &= \lim_{n\to\infty} \frac{\log n}{\log n / \log e} \\ &= \lim_{n\to\infty} \frac{\log e \log n}{\log n} \\ &= \log e\end{aligned}$$

As described above, $1 < \log e < 2$.

S-Problems

Problem S1. Yes, the sets $\{1, 1, 2\}, \{1, 2\}, \{2, 1\}$ are all equal. $\{1, 1, 2\} = \{1, 2\}$ because these sets have the same elements, namely, 1, 2, and two sets are equal if and only if they have the same elements (duplicates are irrelevant). Similarly, $\{1, 2\} = \{2, 1\}$ because these sets have the same elements (order is irrelevant).

Problem S2. The statement

$$\{1, \{2,3\}\} \subseteq \{1, 2, 3, 4, 5, \ldots\}$$

is false. The first set contains an element that is *not* an element of the second set—namely, $\{2, 3\}$.

Problem S3. $\mathcal{P}(\{1, 2\}) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$.

Problem S4. The permutations of the set $\{1, 3, 4\}$ are

$$[1, 3, 4], [1, 4, 3], [3, 1, 4], [3, 4, 1], [4, 1, 3], [4, 3, 1].$$

Problem S5. We are arranging 5 students out of 9 students in chairs; therefore, this is a permutations problem: compute $P_{9,5}$.

$$P_{9,5} = \frac{9!}{(9-5)!} = 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5.$$

Problem S6. A committee of 3 representatives is to be chosen from a larger group of 20 people. The number of ways this committee can be formed is $C(20, 3) = \frac{20 \cdot 19 \cdot 18}{3 \cdot 2 \cdot 1}$.

DGF-Problems.

Problem DGF1. First we verify that f is 1-1. This is done by showing that if $k \neq m$ are distinct elements of the domain of f (in this case, elements of $\mathbf{N}$) then $f(k) \neq f(m)$. In other words, we need to show that $k^2 \neq m^2$—but this is obvious since $k \neq m$.

Next we determine the range of f. Notice that each output of f is a square, a number of the form m^2. But the squares are $0, 1, 4, 9, 16, 25, \ldots$. Therefore, the range of f is the set

$$\{0, 1, 4, 9, 16, \ldots\} = \{r \mid \text{for some } m, r = m^2\}$$

Finally, we see that, since many natural numbers are not squares, f is not onto. In particular, notice that 2 is an element of the codomain $\mathbf{N}$ of f that is not one of the output values of f—there is no $m \in \mathbf{N}$ for which $m \to 2$ in f.

Problem DGF2. We show that the function $f(n) = n^2$, with domain $\mathbf{N}$, is increasing. Suppose $0 \leq m < n$. We must show $m^2 < n^2$. We have

$$\begin{aligned} m < n \quad &\Rightarrow \quad m \cdot m < n \cdot m \\ &\Rightarrow \quad m^2 < n \cdot n = n^2 \end{aligned}$$

SUM-Problems.

Problem SUM1.

$$\begin{aligned}
\sum_{i=1}^{N} 2i^2 + 3i - 4 &= 2\sum_{i=1}^{N} i^2 + 3\sum_{i=1}^{N} i + 4\sum_{i=1}^{N} 1 \\
&= 2 \cdot \frac{N(N+1)(2N+1)}{6} + 3 \cdot \frac{N(N+1)}{2} + 4N \\
&= \frac{N}{6} \cdot (4N^2 + 15N + 35) \\
&= \frac{1}{6} \cdot (4N^3 + 15N^2 + 35N)
\end{aligned}$$

MI-Problems.

Problem MI1. The statement $\phi(n)$ to be established for all $n \geq 1$ is:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

For the Basis Step, notice that $\phi(1)$ is the statement

$$\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$$

which is obviously true. For the Induction Step, we assume $\phi(n)$ is true, and we prove $\phi(n+1)$. $\phi(n+1)$ is the following statement:

$$\sum_{i=1}^{n+1} i = \frac{(n+1)(n+2)}{2}$$

To prove $\phi(n+1)$ is true, we follow these steps:

$$\begin{aligned}
\sum_{i=1}^{n+1} i &= \Big(\sum_{i=1}^{n} i\Big) + (n+1) \\
&= \frac{n(n+1)}{2} + (n+1) \qquad \text{(by Induction Hypothesis)} \\
&= \frac{n(n+1)}{2} + \frac{2(n+1)}{2} \\
&= \frac{(n+1)(n+2)}{2}
\end{aligned}$$

Problem MI2. Let $\phi(n)$ be the statement

$$n^2 < 2^n.$$

We prove $\phi(n)$ for all $n \geq 5$. Clearly $\phi(5)$ is true because $5^2 < 2^5$.

For the Induction Step, assume $n^2 < 2^n$; we prove $(n+1)^2 < 2^{n+1}$. As a preliminary, let's observe that

$$(*) \qquad n(n-3) > 0, \ (n > 3)$$

which follows from basic algebra (note that for $0 < n < 3$ the expression is < 0 and for $n > 3$, the expression is > 0). From $(*)$ we may conclude

$$(**) \qquad n^2 > 3n > 2n + 1, \ (n > 3)$$

(note that $3n > 2n+1$ whenever $n > 1$). Therefore, we can use the induction hypothesis to reach the conclusion as follows:

$$\begin{array}{rcll} (n+1)^2 & = & n^2 + 2n + 1 & \\ & < & 2n^2 & \text{(by } (**)\text{)} \\ & < & 2 \cdot 2^n & \text{(by Induction Hypothesis)} \\ & = & 2^{n+1}. & \end{array}$$

We have proved the Induction Step. It follows from Mathematical Induction that, for all $n > 4$, $\phi(n)$ holds, that is, $n^2 < 2^n$.

Problem MI3. We craft a statement $\phi(n)$ for this problem:

$$\phi(n) : \text{for all } m, \text{ if } m < n, \text{ then } f(m) < f(n)$$

Notice that $\phi(0)$ is vacuously true (since there are no m that are < 0) and that $\phi(1)$ holds since $2^0 = 1 < 2 = 2^1$. This takes care of the Basis Step.

For the Induction Step, let $n > 1$; we assume $\phi(k)$ holds true for all $k < n$. We prove $\phi(n)$ is true. Suppose $m < n$. We must show that $2^m < 2^n$. If we can show that $2^{n-1} < 2^n$, we will be done, because by the induction hypothesis, if $m < n - 1$, then $2^m < 2^{n-1}$. To verify $2^{n-1} < 2^n$, we proceed as follows:

$$2^{n-1} < 2 \cdot 2^{n-1} = 2^n.$$

We have, by Total Induction, shown that $\phi(n)$ holds for all n. We can use this to show that f is increasing: Suppose $m < n$. We must show that $f(m) < f(n)$. However, this is guaranteed by $\phi(n)$, and so we are done.

Problem MI4. Clearly `factorial(0)` and `factorial(1)` produce correct outputs. We proceed by finite induction on i, where $2 \leq i \leq n$, where $\phi(i)$ is the following statement:

$$\phi(i)\text{: the value stored in accum after the iteration for } i \text{ has finished is } i!$$

For the Basis Step, we notice that before the loop begins with $i = 2$, the value stored in `accum` is 1. In the loop, this value is multiplied by i. The value in `accum` at the end is $2 = 2!$, as required.

For the Induction Step, assume that $\phi(i)$ holds, for $2 \leq i < n$. This means that at the end of the loop with value i, the value in `accum` is $i!$. When the iterator is incremented to $i+1$, the current value of `accum` is multiplied by $i+1$, which is $i+1!$, and this value is stored in `accum`, as required.

Therefore, we have shown by Finite Induction that $\phi(i)$ holds for $2 \leq i \leq n$. But the value returned by `factorial(n)` is the final value stored in `accum`, which, as we have just shown, must be $n!$.

BNT Problems

Problem BNT1. Suppose a, b are integers not both 0.

(A) Let $S = \{z \mid z$ is positive, having the form $ax + by$ for some $x, y\}$, and let $d = \min S$. Then for some x, y we have $d = ax + by$. We first show that $d|a$ and $d|b$. Suppose $d \not| a$. Then there are q, r with $a = dq + r$ and $0 < r < d$. But now we have

$$r = a - dq = a - (ax + by)q = a(1 - xq) + b(-q).$$

Writing r in this way shows that $r \in S$; but this is impossible because $0 < r < d$ and $d = \min S$. Therefore, in fact, $d|a$. A similar argument shows that $d|b$.

Next we show that $d = \gcd(a, b)$ by showing that $d \geq c$ for any common divisor c of a, b. So, let c be a common divisor of a, b. As above, we write $d = ax + by$. Since $c|a$ and $c|b$, we also have $c|(ax + by)$, and so $c|d$. Therefore $d = \gcd(a, b)$.

(B) Suppose $ax' + by' = c$ and $g = \gcd(a, b)$. By the Hint in (A), g is the smallest possible positive linear combination of a and b; we write $g = ax + by$. Then $g \leq c$. If $g \neq c$, use the Division Algorithm to write $c = gq + r$ where $0 < r < g$. Then

$$r = c - gq = (ax' + by') - (aqx + bqy) = a(x' + qx) + b(y' + qy).$$

But now, since $0 < r < g$, this contradicts the fact that g is the smallest positive linear combination of a, b.

Problem BNT2. By the previous Example, we may write $g = ax + by$ for some integers x, y. Since $d|a$ and $d|b$, there are integers s, t such that $a = ds$ and $b = dt$. Therefore

$$g = ax + by = dsx + dty = d(sx + ty).$$

It follows that $d|g$.

Problem BNT3. We may write

$$\begin{aligned} a &= \lfloor \tfrac{a}{n} \rfloor \cdot n + a \bmod n \\ b &= \lfloor \tfrac{b}{n} \rfloor \cdot n + b \bmod n \end{aligned}$$

Subtracting,

$$(*) \qquad a - b = \big(\lfloor\frac{a}{n}\rfloor \cdot n - \lfloor\frac{b}{n}\rfloor \cdot n\big) + \big(a \bmod n - b \bmod n\big).$$

If $n \mid (a-b)$, then since the expression $\big(\lfloor\frac{a}{n}\rfloor \cdot n - \lfloor\frac{b}{n}\rfloor \cdot n\big)$ in $(*)$ is also divisible by n, it follows that $\big(a \bmod n - b \bmod n\big)$ is divisible by n too. But since both $a \bmod n$ and $b \bmod n$ lie in the range $[0, n-1]$, their absolute difference must also lie in the range $[0, n-1]$. Hence, the only way this difference could be divisible by n is if it equals 0. In other words, we must have that $a \bmod n = b \bmod n$.

Conversely, if $a \bmod n = b \bmod n$, then their difference is 0 and so divisible by n. It follows that the entire right-hand side of the expression in $(*)$ is divisible by n. Therefore, $n|(a-b)$.

Problem BNT4. Recall that $q = \lfloor b/a\rfloor = \lfloor -20/7\rfloor = -3$ and $r = b \bmod a = 1$. Therefore $-20 = 7q + r = 7(-3) + 1$. Computing values q' and r' using Java's approach yields:

$$\begin{aligned} q' &= b/a \text{ (integer division)} = -20/7 = -(20/7) = -2 \\ r' &= b\%\, a = -20\%\, 7 = -(20\%\, 7) = -6 \end{aligned}$$

Therefore, we can write $-20 = 7q' + r' = 7(-2) - 6$.

Problem BNT5. We show that for all nonzero a, b, $\gcd(a,b) \cdot \mathrm{lcm}(a,b) = |ab|$. We first prove the two lemmas.

Proof of Lemma 1. We show that if a, b are nonzero integers and $\ell = \mathrm{lcm}(a,b)$, then the rational number $\frac{ab}{\ell}$ is an integer. Suppose $\frac{ab}{\ell}$ is not an integer; this means that $\ell \not| \; ab$. Use the Division Algorithm to find q, r with

$$ab = \ell q + r \quad \text{and} \quad 0 < r < \ell.$$

Subtracting ℓq from both sides yields

$$r = ab - \ell q.$$

Since ℓ and ab are both multiples of a, r must also be a multiple of a. Since ℓ and ab are both multiples of b, r must also be a multiple of b. Therefore, r is a common multiple of a, b, and yet $0 < r < \ell$ — but this is impossible since $\ell = \mathrm{lcm}(a,b)$. Therefore, $\frac{ab}{\ell}$ is in fact an integer.

Lemma 2. We show that if a, b are nonzero integers and $\ell = \mathrm{lcm}(a,b)$, then the integer $\frac{ab}{\ell}$ divides both a and b. By the previous problem, $k = \frac{ab}{\ell}$ is an integer. Let s, t be integers such that $\ell = as = bt$ (using the definition of ℓ). Then we have

$$\begin{aligned} ks &= s \cdot \frac{ab}{\ell} = s \cdot \frac{ab}{as} = s \cdot \frac{b}{s} = b, \text{ and} \\ kt &= t \cdot \frac{ab}{\ell} = t \cdot \frac{ab}{bt} = t \cdot \frac{a}{t} = a \end{aligned}$$

This shows that $k|a$ and $k|b$, as required.

We now prove that, for all nonzero a, b, $\gcd(a,b) \cdot \mathrm{lcm}(a,b) = |ab|$. Let $g = \gcd(a,b)$ and $\ell = \mathrm{lcm}(a,b)$. Let $\ell' = \frac{|ab|}{g}$ and let $g' = \frac{|ab|}{\ell}$. To complete the proof, it suffices to show that $\ell = \ell'$. Our plan is to show that $\ell \leq \ell'$ and that $g' \leq g$. This is enough because, assuming we have established these two inequalities, we will have:

$$\begin{aligned} g' \leq g \quad &\Rightarrow \quad \frac{|ab|}{\ell} \leq \frac{|ab|}{\ell'} \\ &\Rightarrow \quad \frac{1}{\ell} \leq \frac{1}{\ell'} \\ &\Rightarrow \quad \ell' \leq \ell \end{aligned}$$

In other words, we will have both $\ell \leq \ell'$ and $\ell' \leq \ell$.

Claim 1. $\ell \leq \ell'$

Proof of Claim 1. We will show that ℓ' is a common multiple of a, b; it will then follow that $\ell \leq \ell'$. Notice that

$$\ell' = \frac{mn}{g} = m \cdot \frac{n}{g} = n \cdot \frac{m}{g}.$$

Since $\frac{m}{g}$ and $\frac{n}{g}$ are integers (by definition of g), the displayed equations show that ℓ' is a multiple of both m and n, as required.

Claim 2. $g' \leq g$.

Proof of Claim 2. By the previous Example, g' is a common divisor of a, b. Since g is the *greatest* common divisor of a, b, it follows that $g' \leq g$.