Greatest Common Divisors

Let a and b be integers, not both zero. The largest integer d such that $d|a$ and $d|b$ is called the **greatest common divisor** of a and b. The greatest common divisor of a and b is denoted by $\gcd(a, b)$.

Example 1: What is $\gcd(48, 72)$?
The positive common divisors of 48 and 72 are 1, 2, 3, 4, 6, 8, 12, 16, and 24, so $\gcd(48, 72) = 24$.

Example 2: What is gcd(19, 72) ?
The only positive common divisor of 19 and 72 is 1, so $\gcd(19, 72) = 1$.

UMass Boston

Greatest Common Divisors

Using prime factorizations:

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

where $p_1 < p_2 < \dots < p_n$ and $a_i, b_i \in \mathbb{N}$ for $1 \le i \le n$

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

Example:

a = 60 = $2^2\ 3^1\ 5^1$

b = 54 = $2^1\ 3^3\ 5^0$

gcd(a, b) = $2^1\ 3^1\ 5^0$ = 6

UMass Boston

Relatively Prime Integers

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

Examples:

Are 15 and 28 relatively prime?
Yes, gcd(15, 28) = 1.

Are 55 and 28 relatively prime?
Yes, gcd(55, 28) = 1.

Are 35 and 28 relatively prime?
No, gcd(35, 28) = 7.

Relatively Prime Integers

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

Examples:

Are 15, 17, and 27 pairwise relatively prime?
No, because gcd(15, 27) = 3.

Are 15, 17, and 28 pairwise relatively prime?
Yes!
gcd(15, 17) = 1,
gcd(15, 28) = 1, and
gcd(17, 28) = 1.

Least Common Multiples

Definition: The **least common multiple** of the positive integers a and b is the smallest positive integer that is divisible by both a and b.

We denote the least common multiple of a and b by $lcm(a, b)$.

Examples:

lcm(3, 7) = 21

lcm(4, 6) = 12

lcm(5, 10) = 10

Least Common Multiples

Using prime factorizations:

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

where $p_1 < p_2 < \cdots < p_n$ and $a_i, b_i \in \mathbb{N}$ for $1 \leq i \leq n$

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

Example:

a = 60 = $2^2\ 3^1\ 5^1$

b = 54 = $2^1\ 3^3\ 5^0$

lcm(a, b) = $2^2\ 3^3\ 5^1$ = 4·27·5 = 540

GCD and LCM

a = 60 = 2^2 3^1 5^1

b = 54 = 2^1 3^3 5^0

gcd(a, b) = 2^1 3^1 5^0 = 6

lcm(a, b) = 2^2 3^3 5^1 = 540

Theorem: a·b = gcd(a,b)·lcm(a,b)

Modular Arithmetic

Let a be an integer and m be a positive integer. We denote by $a \ mod \ m$ the remainder when a is divided by m.

Examples:

9 mod 4 = 1

9 mod 3 = 0

9 mod 10 = 9

-13 mod 4 = 3

Congruences

Let a and b be integers and m be a positive integer. We say that a is congruent to b modulo m if m divides $a - b$.

We use the notation $a \equiv b\ (mod\ m)$ to indicate that a is congruent to b modulo m.

In other words: $a \equiv b\ (mod\ m)$ if and only if $a\ mod\ m\ = b\ mod\ m.$

UMass Boston

Congruences

Examples:

Is it true that 46 ≡ 68 (mod 11) ?
Yes, because 11 | (46 – 68).

Is it true that 46 ≡ 68 (mod 22)?
Yes, because 22 | (46 – 68).

For which integers z is it true that $z \equiv 12\ (mod\ 10)$?
It is true for any $z \in \{\dots, -28, -18, -8, 2, 12, 22, 32, \dots\}$

Theorem: Let m be a positive integer. The integers a and b are congruent modulo m if and only if there is an integer k such that $a = b + km$.

In-class proving

UMass Boston

Congruences

Theorem: Let m be a positive integer.
If $a \equiv b(mod\ m)$ and $c \equiv d\ (mod\ m)$, then $a + c \equiv b + d(mod\ m)$ and $ac \equiv bd(mod\ m)$.

Proof:
We know that $a \equiv b\ (mod\ m)$ implies that there is an integer s such that $b = a + sm$
Similarly, $c \equiv d\ (mod\ m)$ implies that there is an integer t such that $d = c + tm$.

Therefore,
$b + d = (a + s \cdot m) + (c + t \cdot m) = (a + c) + m(s + t)$
and
$bd = (a + s \cdot m)(c + t \cdot m) = ac + m(a \cdot t + c \cdot s + s \cdot t \cdot m)$

Hence, $a + c \equiv b + d\ (mod\ m)$ and $ac \equiv bd\ (mod\ m)$

The Euclidean Algorithm

The Euclidean Algorithm finds the **greatest common divisor** of two integers a and b.

For example, if we want to find $\gcd(287, 91)$, we divide 287 (the larger number) by 91 (the smaller one):

287 = 91·3 + 14
$\Rightarrow$ 287 - 91·3 = 14
$\Rightarrow$ 287 + 91·(-3) = 14

We know that for integers a, b and c, if $a \mid b$, then $a \mid bc$ for all integers c.

Therefore, any divisor of 91 is also a divisor of $91 \cdot (-3)$.

The Euclidean Algorithm

287 + 91·(-3) = 14

We also know that for integers a, b and c, if $a|b$ and $a|c$, then $a|(b+c)$.

Therefore, any divisor of 287 and 91 must also be a divisor of 287 + 91·(-3), which is 14.

Consequently, the greatest common divisor of 287 and 91 must be the same as the greatest common divisor of 14 and 91:

$$\gcd(287, 91) = \gcd(14, 91)$$

The Euclidean Algorithm

In the next step, we divide 91 by 14: $91 = 14 \cdot 6 + 7$

This means that $\gcd(14, 91) = \gcd(14, 7)$.

So we divide 14 by 7: $14 = 7 \cdot 2 + 0$

We find that $7 \mid 14$, and thus $\gcd(14, 7) = 7$.

Therefore, $\gcd(287, 91) = 7$.

The Euclidean Algorithm

In pseudocode, the algorithm can be implemented as follows:

```
procedure gcd(a, b: positive integers)
x := a
y := b
while y ≠ 0
begin
   r := x mod y
   x := y
   y := r
end {x is gcd(a, b)}
```


Representations of Integers

Let b be a positive integer greater than 1. Then if n is a positive integer, it can be expressed uniquely in the form:

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

where k is a nonnegative integer, $a_0, a_1, \ldots, a_k$ are nonnegative integers less than b, and $a_k \neq 0$.

Example for $b = 10$:

$859 = 8 \cdot 10^2 + 5 \cdot 10^1 + 9 \cdot 10^0$

Representations of Integers

Example for b=2 (binary expansion):

$$(10110)_2 = 1 \cdot 2^4 + 1 \cdot 2^2 + 1 \cdot 2^1 = (22)_{10}$$

Example for b=16 (hexadecimal expansion):

(we use letters A to F to indicate numbers 10 to 15)

$$(3A0F)_{16} = 3 \cdot 16^3 + 10 \cdot 16^2 + 15 \cdot 16^0 = (14863)_{10}$$

Representations of Integers

How can we construct the base b expansion of an integer n?

First, divide n by b to obtain a quotient q_0 and remainder a_0, that is,

$$n = bq_0 + a_0, \text{ where } 0 \leq a_0 < b.$$

The remainder a_0 is the rightmost digit in the base b expansion of n.

Next, divide q_0 by b to obtain:

$$q_0 = bq_1 + a_1, \text{ where } 0 \leq a_1 < b.$$

a_1 is the second digit from the right in the base b expansion of n.
Continue this process until you obtain a quotient equal to zero.

Representations of Integers

Example: What is the base 8 expansion of $(12345)_{10}$?

First, divide 12345 by 8:
$12345 = 8 \cdot 1543 + 1$

$1543 = 8 \cdot 192 + 7$
$192 = 8 \cdot 24 + 0$
$24 = 8 \cdot 3 + 0$
$3 = 8 \cdot 0 + 3$

The result is: $(12345)_{10} = (30071)_8$.

Representations of Integers

```
procedure base_b_expansion(n, b: positive integers)
q := n
k := 0
while q ≠ 0
begin
     a_k := q mod b
     q := ⌊q/b⌋
     k := k + 1
end
{the base b expansion of n is (a_{k-1} … a_1a_0)_b }
```

Addition of Integers

How do we (humans) add two integers?

	1 1 1	carry
Example:	7583	
	+ 4932	
	12515	

	1 1	carry
Binary expansions:	$(1011)_2$	
	+ $(1010)_2$	
	$(10101)_2$	

Addition of Integers

Let $a = (a_{n-1}a_{n-2} \dots a_1a_0)_2$, $b = (b_{n-1}b_{n-2} \dots b_1b_0)_2$.

How can we algorithmically add these two binary numbers?

First, add their rightmost bits:
$a_0 + b_0 = c_0 \cdot 2 + s_0$,

where s_0 is the rightmost bit in the binary expansion of $a + b$, and c_0 is the carry.

Then, add the next pair of bits and the carry:
$a_1 + b_1 + c_0 = c_1 \cdot 2 + s_1$,

where s_1 is the next bit in the binary expansion of $a + b$, and c_1 is the carry.

Addition of Integers

Continue this process until you obtain c_{n-1}.

The leading bit of the sum is $s_n = c_{n-1}$.

The result is: $a + b = (s_n s_{n-1} \dots s_1 s_0)_2$

UMass Boston

Addition of Integers

Example:
Add a = $(1110)_2$ and b = $(1011)_2$.

$a_0 + b_0 = 0 + 1 = 0 \cdot 2 + 1$, so that $c_0 = 0$ and $s_0 = 1$.

$a_1 + b_1 + c_0 = 1 + 1 + 0 = 1 \cdot 2 + 0$, so $c_1 = 1$ and $s_1 = 0$.

$a_2 + b_2 + c_1 = 1 + 0 + 1 = 1 \cdot 2 + 0$, so $c_2 = 1$ and $s_2 = 0$.

$a_3 + b_3 + c_2 = 1 + 1 + 1 = 1 \cdot 2 + 1$, so $c_3 = 1$ and $s_3 = 1$.

$s_4 = c_3 = 1$.

Therefore, $s = a + b = (11001)_2$.

UMass Boston

Addition of Integers

```
procedure add(a, b: positive integers)
c := 0
for j := 0 to n-1 {larger integer (a or b) has n
digits}
begin
   d := ⌊(a_j + b_j + c)/2⌋
   s_j := a_j + b_j + c − 2d
   c := d
end
s_n := c
{the binary expansion of the sum is (s_n s_n-1…s_1 s_0)_2}
```


UMass Boston

59

Counting

Chapter 6 in the textbook

UMass Boston

60

Basic Counting Principles

Counting problems are of the following kinds:

- "**How many** different 8-letter passwords are there?"
- "**How many** possible ways are there to pick 11 soccer players out of a 20-player team?"
- Most importantly, counting is the basis for computing **probabilities of discrete events**.
- "What is the probability of winning the lottery?"

Basic Counting Principles

The sum rule: If a task can be done in n_1 ways and a second task in n_2 ways, and if these two tasks cannot be done at the same time, then there are n_1+n_2 ways to do either task.

Example:
The department will award a free computer to either a CS student or a CS professor. How many different choices are there, if there are 530 students and 15 professors?

There are 530 + 15 = 545 choices.

Basic Counting Principles

Another example:

A student can choose a computer project from one of three lists. The three lists contain 23, 15, and 19 possible projects, respectively. No project is on more than one list. How many possible projects are there to choose from?

Answer: Since no project is on more than one list, by the sum rule there are 23 + 15 + 19 = 57 ways to choose a project.

Basic Counting Principles

Generalized sum rule:

If we have tasks $T_1, T_2, \ldots, T_m$ that can be done in $n_1, n_2, \ldots, n_m$ ways, respectively, and no two of these tasks can be done at the same time, then there are $n_1 + n_2 + \ldots + n_m$ ways to do one of these tasks.

Basic Counting Principles

The product rule: Suppose that a procedure can be broken down into two successive tasks. If there are n_1 ways to do the first task and n_2 ways to do the second task after the first task has been done, then there are n_1n_2 ways to do the procedure.

Generalized product rule:

If we have a procedure consisting of sequential tasks $T_1, T_2, \ldots, T_m$ that can be done in $n_1, n_2, \ldots, n_m$ ways, respectively, then there are $n_1 n_2 \ldots n_m$ ways to carry out the procedure.

Basic Counting Principles

Example: How many different license plates are there that contain exactly three English letters ?

Solution:

There are 26 possibilities to pick the first letter, then 26 possibilities for the second one, and 26 for the last one.

So, there are $26 \cdot 26 \cdot 26 = \mathbf{17576}$ different license plates.

Basic Counting Principles

Other example:

A new company with just two employees, Sanchez and Patel, rents a floor of a building with 12 offices. How many ways are there to assign different offices to these two employees?

Answer:

There are 12 ways to assign an office to Sanchez, then there are only 11 choices for Patel. By the product rule, there are $11 \cdot 12 = 132$ ways to assign offices to Sanchez and Patel

Basic Counting Principles

Another example:

The chairs of an auditorium are to be labeled with an uppercase English letter followed by a positive integer not exceeding 100. What is the largest number of chairs that can be labeled differently?

Answer:

There are 26 uppercase English letters. Each letter can be followed one of 100 positive integer. By the product rule, the largest number of chairs that can be labeled differently is $26 \cdot 100 = 2600$

Basic Counting Principles

The sum and product rules can also be phrased in terms of **set theory**.

Sum rule: Let $A_1, A_2, \ldots, A_m$ be disjoint sets. Then the number of ways to choose any element from one of these sets is $|A_1 \cup A_2 \cup \ldots \cup A_m| = |A_1| + |A_2| + \ldots + |A_m|$.

Product rule: Let $A_1, A_2, \ldots, A_m$ be finite sets. Then the number of ways to choose one element from each set in the order $A_1, A_2, \ldots, A_m$ is $|A_1 \times A_2 \times \ldots \times A_m| = |A_1|\cdot|A_2|\cdot\ldots\cdot|A_m|$

Inclusion-Exclusion

How many bit strings of length 8 either start with a 1 **or** end with 00?

Task 1: Construct a string of length 8 that starts with a 1.

There is one way to pick the first bit (1),
two ways to pick the second bit (0 or 1),
two ways to pick the third bit (0 or 1),
.
.
.
two ways to pick the eighth bit (0 or 1).

Product rule: Task 1 can be done in $1\cdot 2^7 = 128$ ways.

Inclusion-Exclusion

Task 2: Construct a string of length 8 that ends with 00.

There are two ways to pick the first bit (0 or 1),
two ways to pick the second bit (0 or 1),
.
.
.
two ways to pick the sixth bit (0 or 1),
one way to pick the seventh bit (0), and
one way to pick the eighth bit (0).

Product rule: Task 2 can be done in $2^6 = 64$ ways.

Inclusion-Exclusion

Since there are 128 ways to do Task 1 and 64 ways to do Task 2, does this mean that there are 128 + 64 = 192 binary strings either starting with 1 or ending with 00 ?

No, because here Task 1 and Task 2 can be done **at the same time**.

When we carry out Task 1 and create strings starting with 1, some of these strings end with 00.

Therefore, we sometimes do Tasks 1 and 2 at the same time, so **the sum rule does not apply**.

Inclusion-Exclusion

If we want to use the sum rule in such a case, we have to subtract the cases when Tasks 1 and 2 are done at the same time.

How many cases are there, that is, how many strings start with 1 **and** end with 00?

- There is one way to pick the first bit (1),
- two ways for the second, …, sixth bit (0 or 1),
- one way for the seventh, eighth bit (0).

Product rule: In $2^5 = 32$ cases, Tasks 1 and 2 are carried out at the same time.

Inclusion-Exclusion

Since there are 128 ways to complete Task 1 and 64 ways to complete Task 2, and in 32 of these cases Tasks 1 and Task 2 are completed at the same time, there are:

128 + 64 – 32 = 160 ways to do either task.

In set theory, this corresponds to sets A_1 and A_2 that are **not** disjoint. Then we have:

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

This is called the **principle of inclusion-exclusion.**

Inclusion-Exclusion

Example 1:

How many positive integers not exceeding 1000 are divisible by 7 or 11?

Answer:

Let A be the set of numbers are divisible by 7

Let B the set of numbers are divisible by 11.

Then A ∪ B is the set of numbers divisible by 7 or 11, and A ∩ B is set of numbers are divisible by both 7 and 11.

We knew that $|A| = \left\lfloor \frac{1000}{7} \right\rfloor = 142$, $|B| = \left\lfloor \frac{1000}{11} \right\rfloor = 90$, $|A \cap B| = \left\lfloor \frac{1000}{7 \cdot 11} \right\rfloor = 12$

Therefore: |A| + |B|- |A ∩ B| = 142 – 90 – 12 = 220

Inclusion-Exclusion

Example 2:

Suppose that there are 1807 freshmen at UMB. Of these, 453 are taking a course in computer science(CS), 567 are taking a course in mathematics(MATH), and 299 are taking courses in both (CS) and mathematics. How many are not taking a course either in CS or in MATH?

Answer: Let A, B are the sets of students taking course in CS and MATH respectively. It follows that |A| + |B|- |A ∩ B| = 453 + 567 – 299 = 721

Inclusion-Exclusion

Another example:

A marketing report concerning personal computers states that 650,000 owners will buy a printer for their machines next year and 1,250,000 will buy at least one software package. If the report states that 1,450,000 owners will buy either a printer or at least one software package, how many will buy both a printer and at least one software package?

Answer:

Let A be the set of people will by a printer → |A| = 650K

Let B be the set of people will by at least one software package → |B| = 1250K

The report said that |A| + |B| - |A ∩ B| = 1450K

Therefore: |A ∩ B| = |A| + |B| - 1450K = 1900K – 1450K = 450K

Basic Counting Principles

Division rule: There are $n\ /\ d$ ways to do a task if it can be done using a procedure that can be carried out in n ways, and for every way w, exactly d of the n ways correspond to way w

Example: How many ways are there to seat four people around a circular table, where two seatings are considered the same when each person has the same left neighbor and the same right neighbor?

- We number the seats in clockwise around the table (S1,...,S4)
- There are 4 ways to select the person for S1; 3 ways to select the person for S2; 2 ways to select the person for S3, and one way to select the person for S4. Thus, there are 4! = 24 ways to order the given four people for these seats.
- However, there are 4 ways of rotation the peoples in the same arrangement. Therefore, there are 4!/4 = 6 different ways to arrange for 4 people around the circular table.

Basic Counting Principles

Example: There are 6 blocks (4 red, and 2 white). How many way to arrange these 6 blocks?

Solution:

- There are 6 ways to select a block for 1st position; 5 ways to select a block for 2nd position;… 1 way to select a block for the 6th position.
  Thus, there are $6! = 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$
- However, in the 4 red blocks we have 4! arrangements are considered identical. Similarly, in 2 white blocks, we have 2 arrangements are considered identical.
- Therefore, by the division rule, the ways to arrange these 6 blocks are $\frac{6!}{4!2!} = 15$

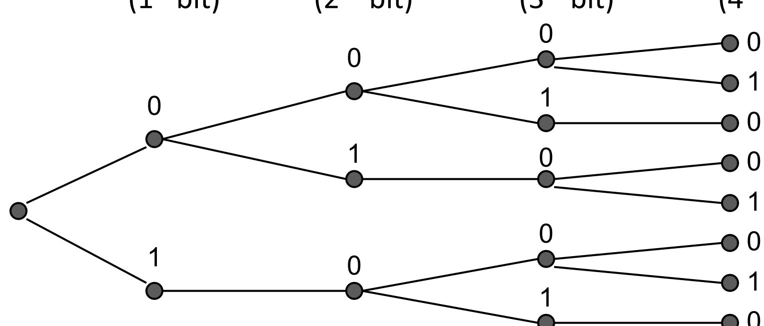
UMass Boston

79

Tree Diagrams

How many bit strings of length four do not have two consecutive 1s?

Task 1	Task 2	Task 3	Task 4
(1st bit)	(2nd bit)	(3rd bit)	(4th bit)



There are 8 strings.

UMass Boston

80

The Pigeonhole Principle

The pigeonhole principle: If (k + 1) or more objects are placed into k boxes, then there is at least one box containing two or more of the objects.

Example 1: If there are 11 players in a soccer team that wins 12-0, there must be at least one player in the team who scored at least twice (assuming there are no own goals!).

Example 2: If you have 6 classes from Monday to Friday, there must be at least one day on which you have at least two classes.

Generalized Pigeonhole Principle

If *N* objects are placed into *k* boxes, then there is at least one box containing at least $\lceil N/k \rceil$ objects.

Example: Assume you have a drawer containing a random distribution of a dozen brown socks and a dozen black socks. It is dark, so how many socks do you have to pick to be sure that among them there is a matching pair?

Answer: There are two types of socks, so if you pick 3 socks, there must be either at least two brown socks or at least two black socks.

This applied the generalized pigeonhole principle: $\lceil 3/2 \rceil = 2$.

Generalized Pigeonhole Principle

Another example:

What is the minimum number of students required in a discrete mathematics class to be sure that at least six will receive the same grade, if there are five possible grades, A, B, C, D, and F?

Answer:

The minimum number of students needed to ensure that at least six students receive the same grade is the smallest integer N such that $\lceil N/5 \rceil = 6$.

The smallest such integer is $N = 5 \cdot 5 + 1 = 26$.

Thus, 26 is the minimum number of students needed to ensure that at least six students will receive the same grade.