What's It All About?

- Continuous mathematics—*calculus*—considers objects that vary continuously
  - distance from the wall
- Discrete mathematics considers *discrete* objects, that come in *discrete* bundles
  - number of babies: can't have 1.2

The mathematical techniques for discrete mathematics differ from those for continuous mathematics:

- counting/combinatorics
- graph representations
- probability
- logic

We'll be studying these techniques in this course.

Why is it computer science?

This is basically a mathematics course:

- no programming
- lots of theorems to prove

So why is it computer science?

Discrete mathematics is the mathematics underlying almost all of computer science:

- Designing high-speed networks
- Finding good algorithms for sorting
- Doing good web searches
- Analysis of algorithms
- Proving algorithms correct

This Course

We will be focusing on:

- Tools for discrete mathematics:
  - graphs and trees (Chapter 3)
  - counting/combinatorics (Chapter 4)
  - probability (Chapter 6)
  - logic (Chapter 7)
- Tools for proving things:
  - induction (Chapter 2)
  - (to a lesser extent) recursion
- Algorithms (Chapter 1):
  - finding optimal algorithms for tasks
  - analyzing algorithms
    - recurrence relations (Chapter 5)

Typical Problem: Scheduling

Given:

Task	Time Needed	Constraints
1. Unload luggage	20	
2. Unload passengers	10	
3. Load new luggage	20	after 1
4. Clean the cabin	15	after 2
5. Load on more food	10	after 2
6. Load new passengers	25	after 4

Problem: What's the minimum amount of time we need to perform all these tasks?

Could perform tasks sequentially, but that clearly isn't optimal.

What's the best way to model this? Use a graph!

Analysis:

- Let $T(k)$ = soonest Time we can finish task k.
- Let $d(k)$ = duration of task k
  - E.g., $d(1) = 20$; $d(4) = 15$
- **Key observation:** $T(k) = d_k + \max_{j \to k} T(j)$
  - $T(F) = 0 + \max\{T(5), T(6), T(3)\}$

Graphs

This is a simple problem; maybe you could do it without a graph

- For big problems (the kind that arise in industry) the graph representation is indispensible

Graphs come up everywhere!

- Exactly the same graph is used to figure out whether we can parallelize a program (program dependency graph)
- Modeling ATM networks
- Modeling transportation routes
- Modeling probabilistic dependencies (Bayesian networks)
- ...

The nodes in a graph represent *discrete* objects.

Sets

You need to be comfortable with set notation:

$$S = \{m | 2 \leq m \leq 100, m \text{ is an integer}\}$$

S is
the set of
all m
such that
m is between 2 and 100
and
m is an integer.

Important Sets

(More notation you need to know and love ...)

- N (occasionally $\mathbb{N}$): the nonnegative integers $\{0, 1, 2, 3, \ldots\}$
- N^+: the positive integers $\{1, 2, 3, \ldots\}$
- Z: all integers $\{\ldots, -3, -3, -1, 0, 1, 2, 3, \ldots\}$
- Q: the rational numbers $\{a/b : a, b \in Z, b \neq 0\}$
- R: the real numbers
- Q^+, R^+: the positive rationals/reals

Set Notation

- $|S|$ = *cardinality of* (number of elements in) S
  - $|\{a, b, c\}| = 3$
- **Subset**: $A \subset B$ if every element of A is an element of B
  - Note: Lots of people (including me, but not the authors of the text) usually write $A \subset B$ only if A is a *strict* or *proper* subset of B (i.e., $A \neq B$). I write $A \subseteq B$ if $A = B$ is possible.
- Power set: $\mathcal{P}(S)$ is the set of all subsets of S (sometimes denoted 2^S).
  - E.g., $\mathcal{P}(\{1, 2, 3\}) =$ $\{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$
  - $|\mathcal{P}(S)| = 2^{|S|}$

Set Operations

- **Union:** $S \cup T$ is the set of all elements in S or T
  - $S \cup T = \{x | x \in S \text{ or } x \in T\}$
  - $\{1, 2, 3\} \cup \{3, 4, 5\} = \{1, 2, 3, 4, 5\}$
- **Intersection:** $S \cap T$ is the set of all elements in both S and T
  - $S \cap T = \{x | x \in S, x \in T\}$
  - $\{1, 2, 3\} \cap \{3, 4, 5\} = \{3\}$
- **Set Difference:** $S - T$ is the set of all elements in S not in T
  - $S - T = \{x | x \in S, x \notin T\}$
  - $\{3, 4, 5\} - \{1, 2, 3\} = \{4, 5\}$
- **Complementation:** $\overline{S}$ is the set of elements not in S
  - What is $\overline{\{1, 2, 3\}}$?
  - Complementation doesn't make sense unless there is a *universe*, the set of elements we want to consider.
  - If U is the universe, $\overline{S} = \{x | x \in U, x \notin S\}$
  - $\overline{S} = U - S$.

Venn Diagrams

Sometimes a picture is worth a thousand words (at least if we don't have two many sets involved).

A Connection

Lemma: For all sets S and T, we have

$$S = (S \cap T) \cup (S - T)$$

Proof: We'll show (1) $S \subset (S \cap T) \cup (S - T)$ and (2) $(S \cap T) \cup (S - T) \subset S$.

For (1), suppose $x \in S$. Either
(a) $x \in T$ or (b) $x \notin T$.

If (a) holds, then $x \in S \cap T$.

If (b) holds, then $x \in S - T$.

In either case, $x \in (S \cap T) \cup (S - T)$.

Since this is true for all $x \in S$, we have (1).

For (2), suppose $x \in (S \cap T) \cup (S - T)$. Thus, either (a) $x \in (S \cap T)$ or $x \in (S - T)$. Either way, $x \in S$.

Since this is true for all $x \in (S \cap T) \cup (S - T)$, we have (2).

Two Important Morals

1. One way to show $S = T$ is to show $S \subset T$ and $T \subset S$.
2. One way to show $S \subset T$ is to show that for every $x \in S$, x is also in T.

Relations

- **Cartesian product**:
  $S \times T = \{(s,t) : s \in S, t \in T\}$
  - $\{1,2,3\} \times \{3,4\} =$
    $\{(1,3),(2,3),(3,3),(1,4),(2,4),(3,4)\}$
  - $|S \times T| = |S| \times |T|$.
- A *relation* on S and T (or, on $S \times T$) is a subset of $S \times T$
- A *relation* on S is a subset of $S \times S$
  - *Taller than* is a relation on people: (Joe,Sam) is in the Taller than relation if Joe is Taller than Sam
  - *Larger than* is a relation on R:
    $$L = \{(x,y)|x,y \in R, x > y\}$$
  - *Divisibility* is a relation on N:
    $$D = \{(x,y)|x,y \in N, x|y\}$$

Reflexivity, Symmetry, Transitivity

- A relation R on S is *reflexive* if $(x,x) \in R$ for all $x \in S$.
  - $\leq$ is reflexive; $<$ is not
- A relation R on S is *symmetric* if $(x,y) \in R$ implies $(y,x) \in R$.
  - "sibling-of" is symmetric (what about "sister of")
  - $\leq$ is not symmetric
- A relation R on S is *transitive* if $(x,y) \in R$ and $(y,z) \in R$ implies $(x,z) \in R$.
  - $\leq$, $<$, $\geq$, $>$ are all transitive;
  - "parent-of" is not transitive; "ancestor-of" is

Pictorially, we have:

Transitive Closure

[[NOT DISCUSSED ENOUGH IN THE TEXT]]

The *transitive closure* of a relation R is the least relation R^* such that

1. $R \subset R^*$
2. R^* is transitive (so that if $(u,v),(v,w) \in R^*$, then so is (u,w)).

Example: Suppose $R = \{(1,2),(2,3),(1,4)\}$.

- $R^* = \{(1,2),(1,3),(2,3),(1,4)\}$
- we need to add $(1,3)$, because $(1,2),(2,3) \in R$

Note that we don't need to add (2,4).

- If (2,1), (1,4) were in R, then we'd need (2,4)
- (1,2), (1,4) doesn't force us to add anything (it doesn't fit the "pattern" of transitivity.

Note that if R is already transitive, then $R^* = R$.

Equivalence Relations

- A relation R is an *equivalence relation* if it is reflexive, symmetric, and transitive
  - $=$ is an equivalence relation
  - *Parity* is an equivalence relation on N; $(x, y) \in Parity$ if $x - y$ is even

Functions

We think of a function $f : S \to T$ as providing a mapping from S to T. But ...

Formally, a *function* is a relation R on $S \times T$ such that for each $s \in S$, there is a unique $t \in T$ such that $(s, t) \in R$.

If $f : S \to T$, then S is the *domain* of f, T is the *range*; $\{y : f(x) = y \text{ for some } x \in S\}$ is the *image*.

We often think of a function as being characterized by an algebraic formula

- $y = 3x - 2$ characterizes $f(x) = 3x - 2$.

It ain't necessarily so.

- Some formulas don't characterize functions:
  - $x^2 + y^2 = 1$ defines a circle; no unique y for each x
- Some functions can't be characterized by algebraic formulas
  - $f(n) = \begin{cases} 0 & \text{if } n \text{ is even} \\ 1 & \text{if } n \text{ is odd} \end{cases}$

Function Terminology

Suppose $f : S \to T$

- f is *onto* (or *surjective*) if, for each $t \in T$, there is some $s \in S$ such that $f(s) = t$.
  - if $f : R^+ \to R^+$, $f(x) = x^2$, then f is onto
  - if $f : R \to R$, $f(x) = x^2$, then f is *not* onto
- f is *one-to-one* (1-1, *injective*) if it is not the case that $s \neq s'$ and $f(s) = f(s')$.
  - if $f : R^+ \to R^+$, $f(x) = x^2$, then f is 1-1
  - if $f : R \to R$, $f(x) = x^2$, then f is *not* 1-1.

- a function is *bijective* if it is 1-1 and onto.
  - if $f : R^+ \to R^+$, $f(x) = x^2$, then f is bijective
  - if $f : R \to R$, $f(x) = x^2$, then f is *not* bijective.

If $f : S \to T$ is bijective, then $|S| = |T|$.

Inverse Functions

If $f : S \to T$, then f^{-1} maps an element in the range of f to all the elements that are mapped to it by f.

$$f^{-1}(t) = \{s | f(s) = t\}$$

- if $f(2) = 3$, then $2 \in f^{-1}(3)$.

f^{-1} is not a function from range(f) to S.

It is a function if f is one-to-one.

- In this case, $f^{-1}(f(x)) = x$.

Functions You Should Know (and Love)

- *Absolute value*: Domain = R; Range = $\{0\} \cup R^+$

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

  - $|3| = |-3| = 3$
- *Floor function*: Domain = R; Range = Z

$$\lfloor x \rfloor = \text{ largest integer not greater than } x$$

  - $\lfloor 3.2 \rfloor = 3$; $\lfloor \sqrt{3} \rfloor = 1$; $\lfloor -2.5 \rfloor = -3$
- *Ceiling function*: Domain = R; Range = Z

$$\lceil x \rceil = \text{ smallest integer not less than } x$$

  - $\lceil 3.2 \rceil = 4$; $\lceil \sqrt{3} \rceil = 2$; $\lceil -2.5 \rceil = -2$
- *Factorial function*: Domain = Range = N

$$n! = n(n-1)(n-2)...3 \times 2 \times 1$$

  - $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$
  - By convention, $0! = 1$

Modular arithmetic

[[MOSTLY NOT IN THE TEXT]]

Mod function: Domain = $Z \times N^+$; Range = N

- Informally, m mod n is the remainder after you divide m by n
- $8 \bmod 3 = 2$; $53 \bmod 20 = 13$; $-8 \bmod 3 = 1$
- Two equivalent formal definitions:
  - $n \bmod m = n - \lfloor n/m \rfloor m$
  - $n \bmod m = r$, where $n = qm + r$, $0 \leq r < m$
- The text assumes that n is a positive integer, but this definition makes sense if n is an arbitrary integer

Modular Arithmetic

ALSO NOT IN THE TEXT:

$a \equiv b (\mathrm{mod} m)$ means that a and b are congruent to the same thing modulo m

- $a = q_1 m + r$; $b = q_2 m + r$
- $a \equiv b (\mathrm{mod} m)$ iff $a - b$ is divisible by m
- **Example:** $17 \equiv 27 (\mathrm{mod} 5)$

You can add, subtract, and multiply modulo m:

- $(2 + 6) (\mathrm{mod} 7) = 8 (\mathrm{mod} 7) = 1 (\mathrm{mod} 7)$
- $(2 \times 6) (\mathrm{mod} 7) = 12 (\mathrm{mod} 7) = 5 (\mathrm{mod} 7)$

More precisely, if $a \equiv a' (\mathrm{mod} m)$ and $b \equiv b' (\mathrm{mod} m)$ then

- $a + b \equiv a' + b' (\mathrm{mod} m)$
- $a \times b \equiv a' \times b' (\mathrm{mod} m)$

Hashing

One application of modular arithmetic is *hashing*:

Problem: How to store lots of things in relatively few memory locations (for quick retrieval). For example, you may want to store the records of $200,000,000$ people in $1,000,000$ record locations, to allow for quick retrieval.

- One approach: store the first 200 in memory location 1, the next 200 in memory location 2, etc.
- Problem: if you add one more person to the list, it might throw everything off.
- Better solution: Compute the person's social security number mod $1,000,000$, and store the information in that memory location.