Inductive Definitions

Example: Define $\Sigma_{k=1}^{n} a_k$ inductively (i.e., by induction on n):

- $\Sigma_{k=1}^{1} a_k = a_1$
- $\Sigma_{k=1}^{n+1} a_k = \Sigma_{k=1}^{n} a_k + a_{n+1}$

The inductive definition avoids the use of $\cdots$, and thus is less ambiguous.

Example: An inductive definition of $n!$:

- $1! = 1$
- $(n+1)! = (n+1)n!$

Could even start with $0! = 1$.

Inductive Definitions of Sets

A *palindrome* is an expression that reads the same backwards and forwards:

- Madam I'm Adam
- Able was I ere I saw Elba

What is the set of palindromes over $\{a, b, c, d\}$? Two approaches:

1. The smallest set P such that
   (a) P contains a, b, c, d, aa, bb, cc, dd
   (b) if x is in P, then so is axa, bxb, cxc, and dxd
2. Define P_n, the palindromes of length n, inductively:
   - $P_1 = \{a, b, c, d\}$
   - $P_2 = \{aa, bb, cc, dd\}$
   - $P_{n+1} = \{axa, bxb, cxc, dxd | x \in P_{n-1}\}$, $n \geq 2$

   Let $P' = \cup_n P_n$.

Theorem: $P = P'$. (The two approaches define the same set.)
Proof: Show $P \subseteq P'$ and $P' \subseteq P$.
To see that $P \subseteq P'$, it suffices to show that

(a) P' contains a, b, c, d, aa, bb, cc, dd

(b) if x is in P', then so is axa, bxb, cxc, and dxd

(since P is the least set with these properties).

Clearly $P_1 \cup P_2$ satisfies (1), so P' does. And if $x \in P'$, then $x \in P_n$ for some n, in which case axa, bxb, cxc, and dxd are all in P_{n+2} and hence in P'. Thus, $P \subseteq P'$.

To see that $P' \subseteq P$, we prove by strong induction that $P_n \subseteq P$ for all n. Let $P(n)$ be the statement that $P_n \subseteq P$.

Basis: $P_1, P_2 \subseteq P$: Obvious.

Suppose $P_1, \ldots, P_n \subseteq P$. If $n \geq 2$, the fact that $P_{n+1} \subseteq P$ follows immediately from (b). (Actually, all we need is the fact that $P_{n-1} \subseteq P$, which follows from the (strong) induction hypothesis.)

Thus, $P' = \cup_n P_n \subseteq P$.

Recall that the set of palindromes is the smallest set P such that

(a) P contains a, b, c, d, aa, bb, cc, dd

(b) if x is in P, then so is axa, bxb, cxc, and dxd

"Smallest" is not in terms of cardinality.

- P is guaranteed to be infinite

"Smallest" is in terms of the subset relation.

Here's a set that satisfies (a) and (b) and isn't the smallest:

Define Q_n inductively:

- $Q_1 = \{a, b, c, d\}$
- $Q_2 = \{aa, bb, cc, dd, ab\}$
- $Q_{n+1} = \{axa, bxb, cxc, dxd | x \in Q_{n-1}\}$, $n \geq 2$

Let $Q = \cup_n Q_n$.

It's easy to see that Q satisfies (a) and (b), but it isn't the smallest set to do so.

Just a Reminder

(from your friendly sponsor)

What's (usually) a key step in proving a property of an algorithm:

Find a loop invariant!

- State clearly what the invariant is
- Prove that it holds (often by induction, since the invariant says "On the nth iteration of the loop, property $P(n)$ holds")

Graphs and Trees

Graphs and trees come up everywhere. We saw an example in Chapter 0 of a *precedence graph*. Here's another example of where graphs come in handy:

A farmer is bringing a wolf, a cabbage, and a goat to market. They need to cross a river in a boat which can accommodate only two things, including the farmer. Moreover:

- the farmer can't leave the wolf alone with the goat
- the farmer can't leave the goat alone with the cabbage

How should he cross the river?

Getting a good representation is the key.

What are the allowable configurations?

- A configuration looks like (X, Y), where $X, Y \subseteq \{W, C, F, G\}$
- Can have X on the initial side of the river, Y on the other

$(WCFG, \emptyset)$	$(\emptyset, WCFG)$
(WCF, G)	(G, WCF)
(WGF, C)	(C, WGF)
(CGF, W)	(FG, WC)
(WC, FG)	(W, CFG)

What's the initial configuration?

- $(WCFG, \emptyset)$

Use a graph to represent when we can get from one configuration to another.

Other Examples

Niche graphs (Ecology):

- The vertices are species
- Two vertices are connected by an edge if they compete (use the same food resources, etc.)

Niche graphs give a visual representation of competitiveness.

Influence Graphs

- The vertices are people
- There is an edge from a to b if a influences b

Influence graphs give a visual representation of power structure.

There are lots of other examples in all fields ...

Terminology and Notation

A *graph* G is a pair (V, E), where V is a set of *vertices* or *nodes* and E is a set of *edges* or *branches*; an edge is a set $\{v, v'\}$ of two not necessarily distinct vertices (i.e., $v, v' \in V$).

- We sometimes write $G(V, E)$ instead of G
- If $V = \emptyset$, then $E = \emptyset$, and G is called the *null graph*.

We usually represent a graph pictorially.

- A vertex with no edges incident to it is said to be *isolated*
- If $\{v\} \in E$ (the book writes $\{v, v\}$), then there is a *loop* at v
- $G'(V', E')$ is a *subgraph* of $G(V, E)$ if $V' \subseteq V$ and $E' \subseteq E$.

Directed Graphs

Note that $\{v, u\}$ and $\{u, v\}$ represent the same edge.

In a *directed graph* (*digraph*), the order matters. We denote an edge as (v, v') rather than $\{v, v'\}$. We can identify an undirected graph with the directed graph that has edges (v, v') and (v', v) for every edge $\{v, v'\}$ in the undirected graph.

Two vertices v and v' are *adjacent* if there is an edge between them, i.e., $\{v, v'\} \in E$ in the undirected case, $(v, v') \in E$ or $(v', v) \in E$ in the directed case.

Representing Relations Graphically

Given a relation R on $S \times T$, we can represent it by the directed graph $G(V, E)$, where

- $V = S \cup T$ and
- $E = \{(s, t) : (s, t) \in R\}$

Example: Represent the $<$ relation on $\{1, 2, 3, 4\}$ graphically.

How does the graphical representation show that a graph is

- reflexive?
- symmetric?
- transitive?

Multigraphs

In a *multigraph*, there may be several edges between two vertices.

- There may be several roads between two towns.
- There may be several transformations that can change you from one configuration to another
  - This is particularly important in graphs where edges are labeled

Formally, a multigraph $G(V, E)$ consists of a set V of vertices and a *multiset* E of edges

- The same edge can be in more than once

In this course, all graphs are *simple graphs* (not multigraphs) unless explicitly stated otherwise.

- Most of the results generalize to multigraphs

Degree

In a directed graph $G(V, E)$, the *indegree* of a vertex v is the number of edges coming into it

- $\text{indegree}(v) = |\{v' : (v', v) \in E\}|$

The *outdegree* of v is the number of edges going out of it:

- $\text{outdegree}(v) = |\{v' : (v, v') \in E\}|$

The *degree* of v, denoted $\deg(v)$, is the sum of the indegree and outdegree.

For an undirected graph, it doesn't make sense to talk about indegree and outdegree. The degree of a vertex is the sum of the edges incident to the vertex, except that we double-count all self-loops.

- Why? Because things work out better that way

Theorem: Given a graph $G(V, E)$,

$$2|E| = \sum_{v \in V} \deg(v)$$

Proof: For a directed graph: each edge contributes once to the indegree of some vertex, and once to the outdegree of some vertex. Thus $|E|$ = sum of the indegrees = sum of the outdegrees.

Same argument for an undirected graph without loops. We need to double-count the loops to make this right in general.

Handshaking Theorem

Theorem: The number of people who shake hands with an odd number of people at a party must be even.

Proof: Construct a graph, whose vertices are people at the party, with an edge between two people if they shake hands. The number of people person p shakes hands with is $\deg(p)$. Split the set of all people at the party into two subsets:

- A = those that shake hands with an even number of people
- B= those that shake hands with an odd number of people

$$\sum_p \deg(p) = \sum_{p \in A} \deg(p) + \sum_{p \in B} \deg(p)$$

- We know that $\Sigma_p \deg(p) = 2|E|$ is even.
- $\Sigma_{p \in A} \deg(p)$ is even, because for each $p \in A$, $\deg(p)$ is even.
- Therefore, $\Sigma_{p \in B} \deg(p)$ is even.

- Therefore $|B|$ is even (because for each $p \in B$, $\deg(p)$ is odd, and if $|B|$ were odd, then $\Sigma_{p \in B} \deg(p)$ would be odd).

Paths

Given a graph $G(V, E)$.

- A *path* in G is a sequence of vertices $(v_0, \ldots, v_n)$ such that $\{v_i, v_{i+1}\} \in E$ ((v_i, v_{i+1}) in the directed case).
- If $v_0 = v_n$, the path is a *cycle*
- An *Eulerian* path/cycle is a path/cycle that traverses every every edge in E exactly once
- A *Hamiltonian* path/cycle is a path/cycle that passes through each vertex in V exactly once.
- A graph with no cycles is said to be *acyclic*

Connectivity

- An undirected graph is *connected* if there is for all vertices u, v, $(u \neq v)$ there is a path from u to v.
- A digraph is *strongly connected* if for all vertices u, v $(u \neq v)$ there is a path from u to v and from v to u.
- If a digraph is connected but not strongly connected, it is *weakly connected.*
- A *connected component* of G is a connected subgraph G' which is not the subgraph of any other connected subgraph of G.

Example: We want the graph describing the interconnection network in a parallel computer:

- the vertices are processors
- there is an edge between two nodes if there is a direct link between them.
  - ◦ if links are one-way links, then the graph is directed

We typically want this graph to be connected.

Trees

A *tree* is a digraph such that

(a) with edge directions removed, it is connected and acyclic

(b) every vertex but one, the *root*, has indegree 1

(c) the root has indegree 0

Trees come up everywhere:

- when analyzing games
- representing family relationships

Bipartite Graphs

A graph $G(V, E)$ is *bipartite* if we can partition V into disjoint sets V_1 and V_2 such that all the edges in E joins a vertex in V_1 to one in V_2.

Example: Suppose we want to represent the "is or has been married to" relation on people. Can partition the set V of people into males (V_1) and females (V_2). Edges join two people who are or have been married.

Complete Graphs and Cliques

- An undirected graph $G(V, E)$ is *complete* if it has no loops and for all vertices u v $(u \neq v)$, $\{u, v\} \in E$.
  - How many edges are there in a complete graph with n vertices?

A complete subgraph of a graph is called a *clique*

- The *clique number* of G is the size of the largest clique in G.