COMP2230/COMP6230 Algorithms

Lecture 1

Table of Contents

Key People

- Course Coordinator: Yuqing Lin
- Lecturer: Nathan Van Maastricht
- Tutor: Nathan Van Maastricht

Key Times

- Lectures
  - Wednesday 8:00am-10:00am (GP201)
- Tutorials
  - Tuesday 2:00pm-4pm (HA58)
  - Wednesday 1:00pm-3:00pm (AVG14)
  - Thursday 2:00pm-4:00pm (HE28)
  - Friday 1:00pm-3:00pm (ES238)

Assumed Knowledge

- SENG1120
- MATH1510

Assessment Items

- Quizzes (3% each). Done on blackboard, as many attempts as you like.
- Mid term tests (13% each). Done in the lectures.
- Assignment (25%). Handed out in week 10, due week 12.
- Final exam (40%). Done in the exam period.

Timeline

Week 1	Lectures start	
Week 2	Tutorials start	
Week 3	Quiz 1	3%
Week 4	Midterm Test 1	13%
Week 5		
Week 6	Quiz 2	3%
Week 7		
Week 8	Midterm Test 2	13%
Week 9		
	Midsemster Break	
	Midsemster Break	
Week 10	Assignment	25%
Week 11		
Week 12	Quiz 3	3%
Week 13		
Exam Period	Formal Exam	40%

Algorithms

Definition (Algorithm)

An *algorithm* is a finite list of instructions, often used to solve a set of problems or perform a computation. Algorithms are unambiguous procedures for performing a calculation, processing data, automated reasoning, and many other tasks.

Algorithms

Algorithms typically have the following

- Input
- Output
- Determinism
- Finiteness
- Correctness

Expressing Algorithms

- High level description. Describe in words.
- Implementation description. Program it.
- Formal description. State function and transition function.

Expressing Algorithms Example

Problem

Given an unsorted list of numbers, find the highest number in the set.

High Level Description Example

1. If there are no numbers in the set, then there is no highest number.
2. Assume the first number in the set is the largest in the set.
3. For each remaining number in the set, if this number is larger than the current largest number, consider this to be the largest number in the set.
4. When there are no numbers left in the set, consider the current largest number to be the largest number in the set.

Implementation Description Example

,

```
/**
Algorithm LargestNumber
Input: An array of numbers list.
Output: The largest number in the array list if list is non empty
    and Integer.MIN_VALUE otherwise.
*/

int LargestNumber(int[] list) {
        if(list != null && list.length == 0){
            return Integer.MIN_VALUE;
        }
        int largest = list[0];
        for(int i = 0; i < list.length; i++) {
            if(list[i] > largest) {
                largest = list[i];
            }
        }

        return largest;
}
```

Algorithm Design

1. Understand the problem
2. Design an algorithm
3. Analyse the algorithm
4. Implement the algorithm

Algorithm Design Example

Problem

Given a positive integer n, factor n into primes.

Problem

Given a node n for an AVL tree, insert n into the AVL tree.

Understanding the Problem

Factoring n is a relatively simple concept, we just want to find the unique primes that when we multiple them together, we get n.

On the other hand, an AVL tree is a self balancing binary search tree. It has the properties that the heights of the two child subtrees of any node differ by at most one; and if at any time they differ by more than one, then we need to rebalance to restore this property.

Design an Algorithm

One of the simplest algorithms to find the prime factors of n would be to just test every number smaller than n to see if it is a factor.

Inserting into AVL trees require a much more complex algorithm on the other hand, the standard implementation has about 40-50 lines of code, but it is also fairly natural as well.

Analyse the Algorithm

The algorithm we came up with for find factors, while simple, is also very slow. Turns out division is a fairly hard problem, which is not an obvious issue with the problem of factoring at first.

The natural algorithm you would typically first try for inserting into an AVL tree is fairly good though. So despite it initially seeming like a much harder problem, once actually trying to understand it, it turns out it's not too hard.

Implement the Algorithm

And finally, once we're happy with our algorithms, we should actually implement them. Despite some algorithms having good theoretical speed, they can be bad in practice. In the same vein algorithms which seem to behave poorly in theory behave really well in practice.

Algorithm Design Summary

Algorithm design at a high level is pretty straight forward, there are four key components. In practice, you will be jumping between the four steps a lot!

Theorems, Lemmas, and Corollaries

Definition (Theorem)

A *theorem* is a statement that has been proven true on the basis of previously established true statements, such as other theorems, lemmas, or corollaries.

Definition (Lemma)

A *lemma* is generally a "helping theorem" which is typically only stated and proved to aid in proving a much more difficult theorem.

Definition (Corollary)

A corollary is a statement that follows with very little proof from another theorem. It is often just a special case of a theorem.

Proof

Definition (Proof)

The proof of a theorem is a logical argument for the theorem statement that follows a deductive list of steps that are justified by other theorems, lemmas, and corollaries.

Example

Is $n^2 - n + 41$ always prime when n is a positive integer?

Exponents

Theorem (Exponent Laws)

Any values in the denominator in the following equations are taken to be non zero.

1. $x^{-n} = \frac{1}{x^n}$.
2. $x^m x^n = x^{m+n}$.
3. $\frac{x^m}{x^n} = x^{m-n}$. *This can be seen by combining the first two exponent laws.*
4. $(x^m)^n = x^{mn}$.
5. $(xy)^m = x^m y^m$.
6. $\left(\frac{x}{y}\right)^m = \frac{x^m}{y^m}$.
7. $\left(\frac{x}{y}\right)^{-m} = \left(\frac{y}{x}\right)^m$. *This can be seen by combining the first and sixth rule.*

Exponent Example

Example (Exponent Example)

To simplify

$$\frac{x^2 x^3}{x^6}$$

we will start by simplifying the numerator using rule 2, that is, add the exponents together, to get

$$\frac{x^5}{x^6}.$$

We are now at a point where we can use rule 3, that is, subtract the exponents, to get

$$x^{-1}.$$

And finally we can use the first rule to get

$$\frac{x^2 x^3}{x^6} = \frac{1}{x}.$$

Logarithms

Theorem (Logarithmic Laws)

Unless otherwise stated, the base of the log does not matter for the following, as long as they are consistent.

1. $\log(1) = 0.$
2. $\log_b(b) = 1.$
3. $\log_b(b^x) = x.$ *This states that the exponential and logarithm cancel each other out.*
4. $\log(xy) = \log(x) + \log(y).$
5. $\log\left(\frac{x}{y}\right) = \log(x) - \log(y)$
6. $\log\left(x^d\right) = d\log(x)$
7. $\log_b(a) = \frac{\log_d(a)}{\log_d(b)}.$ *This law is often called the change of base formula.*

Logarithm Example

Example

We want to simplify

$$\log(x^2) + \log(x^3) - \log(x^6).$$

First we simplify the addition by using rule 4 to give

$$\log(x^2 x^3) - \log(x^6).$$

We now have a subtraction, so we can use rule 5 to give

$$\log\left(\frac{x^2 x^3}{x^6}\right).$$

Logarithm Example Continued

Example

We have

$$\log\left(\frac{x^2x^3}{x^6}\right).$$

Now we can simplify the term we are taking the logarithm of by using our exponential rules, to give

$$\log(x^{-1}),$$

and finally, bringing the -1 out side of the log we will get

$$\log(x^2)+\log(x^3)-\log(x^6)=-\log(x).$$

Sequences

Definition (Finite Sequence)

A finite sequence a is a function from the set $\{1, 2, 3, \ldots, n\}$ to a set X. We denote the ith element in the sequence by a_i or $a[i]$.

Definition (Infinite Sequence)

An infinite sequence a is a function from the set $\{1, 2, 3, \ldots\}$ to a set X. We denote the ith element in the sequence by a_i or $a[i]$.

Limits

Definition (Informal Limit)

A sequence has a limit if the elements of a sequence become closer and closer to some value L, called the limit of the sequence, and they become and remain arbitrarily close to L.

Definition (Formal Limit)

A sequence of real numbers a_n converges to a real number L if, $\forall \varepsilon > 0 \exists N \in \mathbb{N}$ such that $\forall n > N$ we have $|a_n - L| < \varepsilon$.

Limit Laws

Theorem (Limit Laws)

If (a_n) *and* (b_n) *are convergent sequences, the the following limits exist and can be computed using the following.*

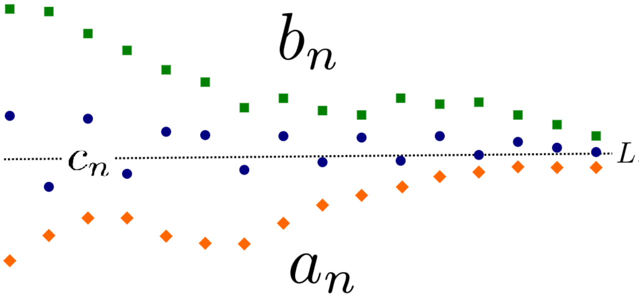
- $\lim\limits_{n\to\infty}(a_n + b_n) = \lim\limits_{n\to\infty} a_n + \lim\limits_{n\to\infty} b_n.$
- $\lim\limits_{n\to\infty} ca_n = c \lim\limits_{n\to\infty} a_n$ *for all constant* $c \in \mathbb{R}$.
- $\lim\limits_{n\to\infty} a_n b_n = \left(\lim\limits_{n\to\infty} a_n\right)\left(\lim\limits_{n\to\infty} b_n\right)$
- $\lim\limits_{n\to\infty} \frac{a_n}{b_n} = \frac{\lim\limits_{n\to\infty} a_n}{\lim\limits_{n\to\infty} b_n}$ *provided that* $\lim\limits_{n\to\infty} b_n \neq 0$

Squeeze Theorem

Theorem (Squeeze Theorem)

If (c_n) is a sequence such that $a_n \leq c_n \leq b_n$ for all $n > N$ and $\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = L$ then

$$\lim_{n\to\infty} c_n = L.$$



Sequence Example

Example

The Fibonacci sequence is a famous infinite sequence

$$1, 1, 2, 3, 5, 8, 13, 21, 34, \ldots$$

which is generated by adding the two previous terms to get the current term. That is $F_n = F_{n-1} + F_{n-2}$.

Increasing/Decreasing Sequences

Definition (Increasing/Decreasing Sequences)

We give the following names to a sequence if they exhibit the corresponding property for all i.

- *Increasing* if $a_i < a_{i+1}$, that is, it only gets bigger.
- *Decreasing* if $a_i > a_{i+1}$, that is, it only gets smaller.
- *Non increasing* if $a_i \geq a_{i+1}$, that is, it never gets bigger, but can stay the same from one term to the next.
- *Non decreasing* if $a_i \leq a_{i+1}$, that is, it never gets smaller, but can stay the same from one term to the next.

Non Decreasing Example

Example

Once again, the Fibonacci sequence is a perfect example. It is a non decreasing sequence as, except for the second term, every term is bigger than the previous, and the second term is the same as the first.

Eventually Increasing/Decreasing

Definition (Eventually Increasing/Decreasing Sequences)

We say a sequence is *eventually* one of the terms from the previous definition if for some $N \in \mathbb{N}$, every term after N satisfies the definition for the term in the previous definition.

Eventually Increasing Example

We can't get away from the Fibonacci sequence, it is eventually increasing because for every term after the second it is an increasing sequence, so it is eventually increasing.

Series

Definition (Series)

A *series* is a sum of an infinite number of terms. We often denote the terms of a series as a_i, and we add together all of the a_i terms. We denote this as

$$\sum_{i=1}^{\infty} a_i.$$

Formally, the infinite series is the limit of the sequence of partial sums of the series, that is

$$\sum_{i=1}^{\infty} = \lim_{n\to\infty} \sum_{i=1}^{n} a_i.$$

Converging series

Definition (Converge)

A series is said to *converge* if the limit of partial sums exist.

How to Think About Series

Think about

$$\sum_{i=1}^{n} a_i$$

as

,

```
int sum = 0;
for(int i = 0; i < n; i++) {
    sum += a[i];
}
```

Geometric series

Definition (Geometric series)

$$S = \sum_{n=0}^{\infty} ar^n = a + ar + ar^2 + ar^3 + ar^4 + \cdots$$

where a is our starting term, r is the common ratio between the two terms.

Example

$$\sum_{i=0}^{\infty} \frac{1}{2}\left(\frac{1}{2}\right)^n = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \cdots$$

Example

$$\sum_{i=0}^{\infty} 3 \cdot 3^n = 3 + 27 + 81 + 243 + \cdots$$

Geometric Series Theorem

Theorem

Given a geometric series

$$S = \sum_{n=0}^{\infty} ar^n,$$

S will converge if and only if $|r| < 1$. *Furthermore, if* $|r| < 1$, *then the sum is given by*

$$S = \frac{a}{1-r}.$$

Geometric Series Example

$$\begin{aligned}
\sum_{n=0}^{\infty} \frac{1}{2}\left(\frac{1}{2}\right)^n &= \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \cdots \\
&= \frac{\frac{1}{2}}{1-\frac{1}{2}} \\
&= \frac{\frac{1}{2}}{\frac{1}{2}} \\
&= 1
\end{aligned}$$

Proofs

Definition (Proof)

The *proof* of a theorem is a logical argument for the theorem statement that follows a deductive list of steps that are justified by other theorems, lemmas, and corollaries.

Types of Proof

- Proof by construction.
- Non constructive proof.
- Proof by contradiction.
- Proof by Induction.

Proof by Construction

Lemma

$2^{99} + 1$ *is a composite number.*

We can approach proving this theorem by finding factors of $2^{99} + 1$, that is, show that it is not prime by constructing the list of primes that combine to make it

Proof.

$$\begin{aligned} 2^{99} + 1 &= \left(2^{33}\right)^3 + 1 \\ &= \left(2^{33} + 1\right)\left(\left(2^{33}\right)^2 - 2^{33} + 1\right) \end{aligned}$$

□

Non Constructive Proof

Lemma

There exist irrational x, y such that x^y is rational.

Proof.

If $\sqrt{2}^{\sqrt{2}}$ is rational, then we are done, take $x = y = \sqrt{2}$.
Otherwise $\sqrt{2}^{\sqrt{2}}$ is irrational, so take $x = \sqrt{2}^{\sqrt{2}}$ and take $y = \sqrt{2}$. Now we have

$$\left(\sqrt{2}^{\sqrt{2}}\right)^{\sqrt{2}}.$$

Using the exponent laws we get

$$\sqrt{2}^{\sqrt{2}\sqrt{2}} = \sqrt{2}^{2} = 2$$

and 2 is rational, so we are done in this case as well. □

Proof by Contradiction

Lemma

$\sqrt{2}$ *is irrational.*

Proof.

Assume that $\sqrt{2} = \frac{a}{b}$ for some integer a and some positive integer b, and the fraction $\frac{a}{b}$ is reduced, that is, a and b have no common factors. Now we can square both sides to get

$$2 = \frac{a^2}{b^2}.$$

Now we can solve this equation for a^2 to get

$$a^2 = 2b^2.$$

Proof by Contradiction Continued

Proof.

$$a^2 = 2b^2$$

shows that 2 is a factor of a^2, so a^2 is even, and it is not hard to show that this means a is even. That is, $a = 2k$ for some k in the integers, so we can rewrite the left hand side of that equation to get

$$(2k)^2 = 2b^2.$$

Expanding and solving for b^2 gives

$$b^2 = 2k^2,$$

which by the same reasoning as earlier means b is even, that is, $b = 2m$ for some m in the positive integers.

Proof by Contradiction Continued

Proof.

So, both a and b have a factor of 2, but we assumed that a and b had no common factors, so this is a contradiction, so our original assumption must be wrong, and we can't write $\sqrt{2}$ as a reduced fraction, so it can't be written as any other fraction either, so it is not rational. $\square$

Proof by Induction

Lemma

$$\sum_{i=1}^{n} i = 1+2+3+4+5+6+7+\cdots+n = \frac{n(n+1)}{2}$$

Proof.

First we show our base case is true. Set $n = 1$ to get the left hand side to be 1, and the right hand side is

$$\frac{1(1+1)}{2} = 1,$$

so the left hand side equals the right hand side, so our base case is true.

Proof by Induction Continued

Proof.

Now we wish to do the inductive component of the proof. We will assume that when $n = k$ we have

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

then we will use that in the next part. We now wish to show that given the previous formula is true, we can show that

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

which is what we would expect if we substituted $n = k + 1$ into the formula of the theorem.

Proof by Induction Continued

Proof.

To show this, we start with our left hand side, so we are only working with

$$\sum_{i=1}^{k+1} i.$$

Expanding this out we get

$$\sum_{i=1}^{k+1} i = 1 + 2 + 3 + \cdots + k + (k+1).$$

The brackets around the last $k+1$ are purely for grouping purposes, the entire contents of the brackets came from the same term, so it helps to see where each component came from in this case. Now we can group this expression in a clever way to use our inductive assumption.

Proof by Induction Continued

Proof.

If we group the terms as follows

$$(1 + 2 + 3 + \cdots + k) + (k + 1)$$

the first set of brackets are exactly the case where $n = k$, so we can replace the expression in the first set of brackets by

$$\frac{k(k+1)}{2}$$

to get

$$\frac{k(k+1)}{2} + (k+1).$$

Now we wish to simplify this expression to hopefully get what we expect.

Proof by Induction Continued

Proof.

First, put both terms over a common denominator of 2, so we get

$$\frac{k(k+1)}{2} + \frac{2(k+1)}{2}.$$

Now we add together these fractions to get

$$\frac{k(k+1) + 2(k+1)}{2}.$$

Finally, factor out a $k+1$ from the numerator to get

$$\frac{(k+1)(k+2)}{2},$$

which is exactly what our expected right hand side was. □

Polynomials

Definition (Polynomial)

A *polynomial* is an expression consisting of variables and coefficients that involve only the operations of addition, subtraction, multiplication and non negative integer exponents of variables.

Example

$x^2 - 4x + 7$ is a polynomial.

Example

$x^{-1} + 3 + x + 3x^2$ is not a polynomial.

Polynomial Laws

Theorem (Polynomial Laws)

If $p(x)$ of degree m and $q(x)$ is a polynomial of degree n, then the following holds.

- $p(x) + q(x)$ *is a polynomial of degree* $\max(\deg(p), \deg(q))$.
- $p(x) - q(x)$ *is a polynomial of degree* $\max(\deg(p), \deg(q))$.
- $p(x)q(x)$ *is a polynomial of degree* $\deg(p) + \deg(q)$.

Logic Notation

Definition (Boolean variable)

A boolean variable p can be either *true* or *false*

We can combine Boolean variables with `OR` ($\vee$), `AND` ($\wedge$), and we can negate with `NOT` ($\neg$). These are often called the *basic operations*.

p	q	$p \vee q$	$p \wedge q$
0	0	0	0
1	0	1	0
0	1	1	0
1	1	1	1

Binomial Coefficients

Definition (Binomial Coefficient)

The binomial coefficient is denoted as $\binom{n}{k}$ and is defined to be

$$\binom{n}{k} = \frac{n!}{(n-k)!k!}.$$

Binomial Coefficient Example

Example

Lets try to calculate $\binom{10}{5}$ without the aid of a calculator or computer.

$$\binom{10}{5} = \frac{10!}{(10-5)!5!}$$

First, lets simplify the brackets in the denominator to get

$$\frac{10!}{5!5!}.$$

Binomial Coefficient Example Continued

Example

Now we will expand out the factorials to see what we can cancel from the numerator and denominator.

$$\frac{10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1}{5 \times 4 \times 3 \times 2 \times 1 \times 5 \times 4 \times 3 \times 2 \times 1}$$

We can cancel off $5 \times 4 \times 3 \times 2 \times 1$ in both the numerator and denominator to get

$$\frac{10 \times 9 \times 8 \times 7 \times 6}{5 \times 4 \times 3 \times 2 \times 1}.$$

Binomial Coefficient Example Continued

Example

Now we can pair the 10 in the numerator and the 5 in the denominator and cancel those to get a 2 in the numerator and a 1 in the denominator, likewise with the 8 and 4, and 6 and 3, so we get

$$\frac{2 \times 9 \times 2 \times 7 \times 2}{1 \times 1 \times 1 \times 2 \times 1}.$$

Binomial Coefficient Example Continued

Example

Simplifying the denominator we get

$$\frac{2 \times 9 \times 2 \times 7 \times 2}{2}.$$

Finally, we can cancel the 2 in the denominator with a 2 from the numerator to be left with

$$9 \times 2 \times 7 \times 2 = 252,$$

so

$$\binom{10}{5} = 252.$$

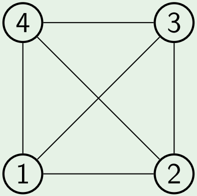
Graphs

Definition (Undirected Graph)

An undirected graph $G = (V, E)$ consists of a set V of vertices, and a set E of edges. An edge $e \in E$ is an *unordered* pair of vertices.

Example

The following is a graph, it has four vertices and six edges. The vertex set is $V = \{1, 2, 3, 4\}$ and the edge set is
$E = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}$.

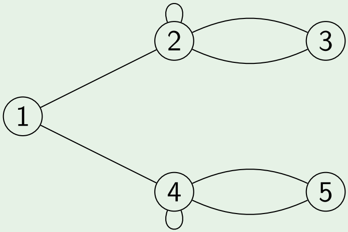


Simple Graphs

A graph without a vertex that is connected to itself, that is, doesn't contain the edge (a, a) for a vertex a, and doesn't have multiple copies of the same edge is called a *simple* graph.

Example

For example, the following graph is not simple because it contains a loop on vertex 2 and 4, and the edge $(2, 3)$ and $(4, 5)$ both occur twice.



Vertex Degree

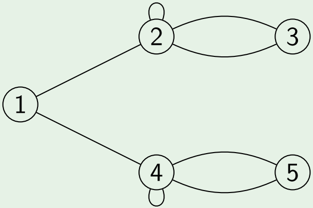
Definition (Vertex Degree)

The *degree* of a vertex v is the number of edges incident on v.

Example

The degree of the vertices in the following graph are

Vertex	1	2	3	4	5
Degree	2	5	2	5	2



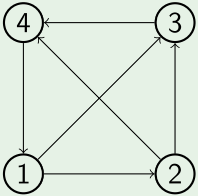
Directed Graphs

Definition (Directed graphs)

A directed graph G consists of a set of vertices V and a set of directed edges E where each $e \in E$ is an *ordered* pair of vertices.

Example

The following is a graph, it has four vertices and six edges. The vertex set is $V = \{1, 2, 3, 4\}$ and the edge set is
$E = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}$.



Adjacency Matrix

Definition (Adjacency Matrix)

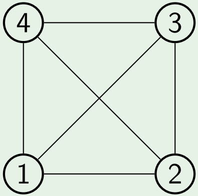
An *adjacency matrix* is a square $|V| \times |V|$ matrix A, such that A_{ij} is 1 if there is an edge from vertex i to vertex j and a 0 if there is no edge.

Adjacency Matrix Example

Example

The adjacency matrix for the following graph is

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

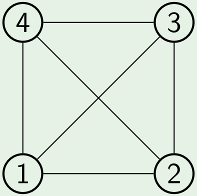


Complete Graph

Definition (Complete Graph)

A *complete* graph is a simple graph where every vertex is connected to every other vertex. The complete graph with n vertices is denoted as K_n.

Example



Bipartite Graphs

Definition (Bipartite Graph)

A bipartite graph is a graph whose vertices can be divided in two disjoint sets U and V, such that every edge connects a vertex from U to a vertex in V. There are no edges from a vertex in U to a vertex in U, likewise there are no edges from a vertex in V to a vertex in V.

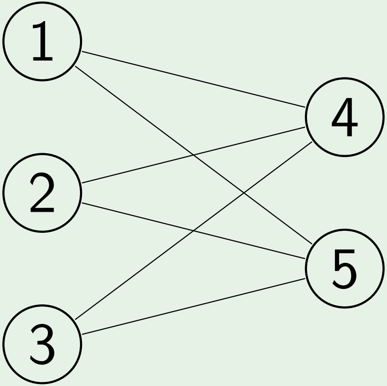
Definition (Complete Bipartite Graph)

A *complete bipartite* graph is a bipartite graph that has an edge from every vertex in U to every vertex in V. The complete bipartite graph where $|U| = m$ and $|V| = n$ is denoted as $K_{m,n}$.

Bipartite Graph Example

Example

The following is the complete bipartite graph $K_{3,2}$. $U = \{1, 2, 3\}$ and the set V is $V = \{4, 5\}$.



Path

Definition (Path)

A *path* of length n from a vertex v_0 to a vertex v_n in a graph G is an alternating sequence of $n+1$ vertices and n edges, starting with v_0 and ending with v_n. When writing paths from simple graphs it is typical to leave out the edges and only list the vertices as there is only one path available to go from vertex u to vertex v.

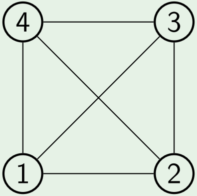
Definition (Simple Path)

A *simple* path from u to v is a path from u to v with no repeated vertices.

Simple Path Example

Example

$(1, 2, 4)$ is a simple path of length 2 in the following graph.



Cycles

Definition (Cycle)

A *cycle* is a path of length greater than 0 from u to u with no repeated edges.

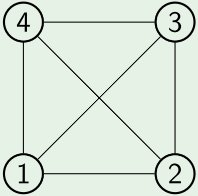
Definition (Simple cycle)

A *simple* cycle is a cycle without repeated vertices except for the beginning and ending vertex.

Cycle Example

Example

$(1, 2, 3, 1)$ is a simple cycle in the following graph.



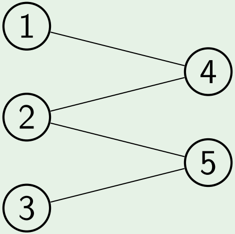
Acyclic Graph

Definition (Acyclic graph)

A graph without any cycles is called *acyclic*

Example

The following is an acyclic graph.

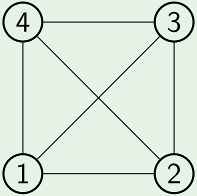


Connected Graph

Definition (Connected Graph)

A graph is *connected* if there exists a path between u and v for every u and v in V

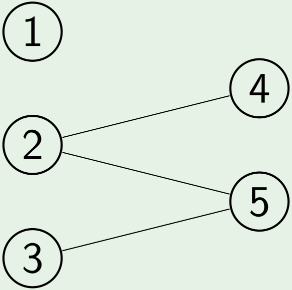
Example



Disconnected Graph

Example

The following graph is not a connected graph, as there is no path from vertex 1 to any other vertex.



Euler Cycles

Definition (Euler Cycle)

An *Euler* cycle in a graph G is a path from vertex u to itself with no repeated edges that contain all the edges and all of the vertices of G.

We know exactly when a graph has a Euler cycle from the following theorem.

Theorem (Eulerian Graph)

A graph G has an Euler cycle if and only if G is connected and the degree of every vertex is even.

Hamiltonian Cycles

Definition (Hamiltonian cycle)

A *Hamiltonian* cycle in a graph G is a cycle that contains each vertex in G exactly once.

Distance Between Vertices

Definition (Distance)

The distance between a vertex u and a vertex v is the shortest path from u to v if it exists, and ∞ if there is no path from u to v. We denote the distance from u to v by $\text{dist}(u, v)$.

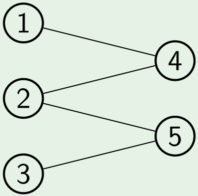
Trees

Definition (Tree)

A tree T is a simple graph such that for any two vertices u and v in T there is a unique simple path from u to v.

Example

The following graph is a tree as there is only a single path from any vertex to any other vertex.



Various Tree Definitions

Theorem

The following statements about a graph G with n vertices are equivalent.

- T *is a tree.*
- T *is connected and acyclic.*
- T *is connected and has* $n-1$ *edges.*
- T *is acyclic and has* $n-1$ *edges.*

Definitions Related to Trees

Definition (Rooted Tree)

In a *rooted* tree a particular vertex is designated as a root.

Definition (Depth of a Vertex in a Rooted Tree)

The *Depth* of a vertex v in a rooted tree is the length of a simple path from the root to v.

Definition (Height of a rooted tree)

The height of a rooted tree is the maximum depth that exists in the tree.

Definitions Related to Trees

Definition (Parent Vertex)

In a rooted tree, the *parent* vertex of a vertex v is the vertex connected to v on the path to the root. Every vertex has a unique parent, except the root which has no parent.

Definition (Child Vertex)

In a rooted tree, a *child* of vertex v is a vertex of which v is the parent.

Definitions Related to Trees

Definition (Leaf Vertex)

A *leaf* is a vertex with no children.

Definition (Internal Vertex)

An *internal* vertex is a vertex that is not a leaf.

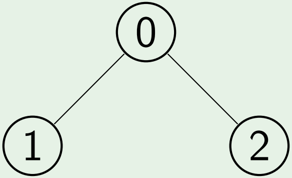
Definition (Binary Tree)

A *binary* tree is a tree such that the maximum number of children of any vertex is 2.

Tree Example

Example

The following is a rooted binary tree, where vertex 0 is the root, 1 and 2 are children of 0, 0 is the parent of 1 and 2. The only internal vertex is 0. 1 and 2 are both leaf vertices. The height of the tree is 1, and vertex 1 and 2 both have depth 1.



Recurrence Relation

Definition (Recurrence Relation)

A *recurrence relation* is a sequence $a_0, a_1, \ldots$ that has an associated equation for generating a_n from earlier terms in the sequence. That is

$$a_n = f(a_{n-1}, a_{n-2}, \ldots, a_0),$$

for some function f.

Definition (Initial Conditions)

Initial conditions for the sequence $a_0, a_1, \ldots$ are explicitly given values for a finite number of terms in the sequence.

Recurrence Relation Example

Example

The classic example is the Fibonacci sequence.

$$f_n = f_{n-1} + f_{n-2}$$

with initial conditions $f_1 = 1$ and $f_2 = 1$.
This produces the sequence $1, 1, 2, 3, 5, 8, 13, 21, \ldots$.

Linear Recurrence Relation

Definition (Linear Recurrence Relation)

A *linear* recurrence relation is a recurrence relation where the function f is of the form

$$\alpha + \sum_{k=1}^{n} c_k a_k$$

where α is a real number, c_k are real numbers, and a_k are terms in the sequence.

Example

$a_n = a_{n-1} + 2a_{n-2} + 3$ is linear as there are only sums of constants multiplied by previous terms and a constant without any previous terms.
$b_n = a_{n-1}a_{n-2}$ is *not* linear there are two previous terms multiplied together.

Homogeneous Recurrence Relations

Definition (Homogeneous Recurrence Relations)

A *homogeneous* recurrence relation is a recurrence relation where the function f is composed of only terms that involve a_k terms.

Example

$a_n = a_{n-1} + 2a_{n-2}$ is homogeneous as every term contains an a_k term.
$a_n = a_{n-1} + 2a_{n-2} + 3$ is *nonhomogeneous* as the term 3 does not contain an a_k term.

Degree of a Recurrence Relation

Definition (Degree of a Recurrence Relation)

The *degree* of a recurrence relation is the largest difference between indices of the terms in the recurrence relation.

Example

The degree of $a_n = a_{n-1} + a_{n-2}$ is 2 as n is the largest index, and $n - 2$ is the smallest, and $n - (n - 2) = n - n + 2 = 2$.

Characteristic Equation

Definition (Characteristic Polynomial)

The *characteristic polynomial* of an associated linear, homogeneous recurrence relation, $a_n = c_{n-1}a_{n-1} + \cdots + c_{n-k}a_{n-k}$ is

$$r^n = c_{n-1}r^{n-1} + \cdots + c_{n-k}r^{n-k}$$

which we would typically rearrange to

$$r^n - c_{n-1}r^{n-1} - \cdots - c_{n-k}r^{n-k} = 0.$$

Example

The characteristic polynomial for $a_n = a_{n-1} + 2a_{n-2}$ is

$$r^2 - r - 2 = 0.$$

Solving Linear, Homogeneous Recurrence Relations

Theorem (Solving Linear, Homogeneous Recurrence Relations)

Given a linear, homogeneous recurrence relation, $a_n = f(a_{n-1}, \ldots, a_{n-k})$*, construct the associated characteristic polynomial, then we can write*

$$a_n = \sum_{i=1}^{k} \sum_{j=0}^{m_i - 1} c_{ij} n^j r_i^n,$$

where c_{ij} *is a constant which can be found using initial conditions,* r_i *is the* i*th root of the characteristic polynomial,* m_i *is the multiplicity of the root* r_i*.*

Distinct Root Recurrence Relation

Example

First, lets look at a recurrence relation that has two distinct roots.

$$a_n = 3a_{n-1} - 2a_{n-2}$$

with initial conditions $a_1 = 2$ and $a_2 = 3$
The associated characteristic polynomial is then

$$r^2 = 3r - 2$$

which can be rearranged to

$$r^2 - 3r + 2 = 0.$$

Now we wish to factor this polynomial to find the roots.

Distinct Root Recurrence Relation

Example

Because this is a quadratic we can use the quadratic formula to find the roots are $r_1 = 1$ and $r_2 = 2$.

Now we just have to use the double sum from the theorem. Seeing as the multiplicity of both roots is 1 our $m_i = 1$ in both cases, so $m_i - 1 = 0$ in both cases, so the second sum is only looked at when $j = 0$.
So we have

$$a_n = c_{1,0} n^0 r_1^n + c_{2,0} n^0 r_2^n.$$

Any number to the power of 0 is just 1, so both the n^0 terms simplify to 1, seeing as there are only two c terms, lets label them nicer and just call them c_1 and c_2, and then plug in the values for our roots to get

$$a_n = c_1 1^n + c_2 2^n.$$

Distinct Root Recurrence Relation

Example

We can further simplify the 1^n because that will always be 1 to get

$$a_n = c_1 + c_2 2^n.$$

This is the general solution to our recurrence relation

Distinct Root Recurrence Relation

Example

Now we use the initial conditions of $a_1 = 2$ and $a_2 = 3$ to find c_1 and c_2 by setting up two simultaneous equations, one when $n = 1$ and the other when $n = 2$ to get

$$2 = c_1 + c_2 2^1$$
$$3 = c_1 + c_2 2^2$$

Solving these gives $c_1 = 1$ and $c_2 = \frac{1}{2}$, so our recurrence relation is

$$a_n = 1 + \frac{1}{2} 2^n$$

which we can simplify to

$$a_n = 1 + 2^{n-1}.$$

After we find the solution using the particular initial conditions, this is called the *particular solution*.

Repeated Root Recurrence Relation

Example

We will find the general solution to the following recurrence relation.

$$a_n = 7a_{n-1} - 16a_{n-2} + 12a_{n-3}.$$

We will approach this problem the same as the last, despite it being a recurrence relation of order 3.

Repeated Root Recurrence Relation

Example

The associated characteristic polynomial is

$$r^3 - 7r^2 + 16r - 12 = 0.$$

This is a cubic, which is not as simple to solve, so we will take for granted that it factors to

$$r^3 - 7r^2 + 16r - 12 = (x-3)(x-2)^2,$$

which you can check by expanding the right hand side. So the roots of the characteristic equation are $r_1 = 3$ and $r_2 = 2$, and r_2 has multiplicity 2.

Repeated Root Recurrence Relation

Example

Now we have to use the double sum from theorem 98. So $m_1 = 1$, so $m_1 - 1 = 0$ so the internal sum is only ran when $j = 0$. But $m_2 = 2$, so $m_2 - 1 = 1$, hence the internal sum is ran when $j = 0$ and $j = 1$. So our nested sums becomes

$$a_n = c_{1,0}n^0 3^n + c_{2,0}n^0 2^n + c_{2,1}n^1 2^n.$$

Simplifying the n^0 terms again gives

$$a_n = c_{1,0}3^n + c_{2,0}2^n + c_{2,1}n2^n.$$

This is our general solution to the recurrence relation. To find a particular solution to it would require three initial conditions, and we would have to set up three simultaneous equations and solve them.