Discrete Structures for Computer Science

George Voutsadakis[1]

[1]Mathematics and Computer Science
Lake Superior State University

LSSU CSci 341

Subsection 1

Analysis of Algorithms

The Optimal Algorithm Problem

- **The Optimal Algorithm Problem**

  Suppose algorithm A solves problem P.
  Is A the best solution to P?
- For us *best* will mean taking the least time.
- Time will be measured by the number of operations of a certain type performed by the algorithm.
  - For a numerical problem we may count, e.g., additions and multiplications.
  - For a sorting problem we may count, e.g., comparisons.
- Since the number of operations depends on the size of the input, we also have to "standardize" measurements with regard to the size of the input.

Worst-Case Complexity

- An input of size n is a **worst case input** if, when compared to all other inputs of size n, it causes A to execute the largest number of operations.
- Consider an input I for an algorithm A.
  - size(I) denotes its size;
  - time(I) denotes the number of operations executed by A on I.
- The **worst case function** for A is defined by

$$W_A(n) = \max\{\text{time}(I) : I \text{ is an input and size}(I) = n\}.$$

- If A and B are algorithms that solve P and if $W_A(n) \leq W_B(n)$ for all $n > 0$, then we know algorithm A has worst case performance that is better than or equal to that of algorithm B.
- An algorithm A is **optimal in the worst case** for problem P if, for any possible algorithm B, $W_A(n) \leq W_B(n)$ for all $n > 0$.

Finding Optimal Algorithms

- Find an algorithm that is optimal in the worst case for a problem P involves, in general three steps:
  1. (Find an algorithm) Find or design an algorithm A to solve P. Then do an analysis of A to find the worst case function W_A.
  2. (Find a lower bound) Find a function F such that $F(n) \leq W_B(n)$ for all $n > 0$ and for all algorithms B that solve P.
  3. Compare F and W_A. If $F = W_A$, then A is optimal in the worst case.
- If we discover that $F \neq W_A$ (i.e., $F(n) < W_A(n)$ for some n) there are two possible courses of action to consider:
  - Try to find a better algorithm than A, i.e., an algorithm C such that $W_C(n) < W_A(n)$ for all $n > 0$.
  - Try to find a "better" lower bound than F, i.e., a new function G, such that $F(n) < G(n) \leq W_B(n)$ for all $n > 0$ and for all algorithms B that solve P.

Example: Finding the Minimum in a List

- We look at an example of an optimal algorithm to find the minimum number in an unsorted list of n numbers.
- We count the number of comparison operations that an algorithm makes between elements of the list.
- To find the minimum number in a list of n numbers, the minimum number must be compared with the other $n-1$ numbers.

  So $n-1$ is a lower bound on the number of comparisons needed to find the minimum number in a list of n numbers.
- If we represent the list as an array a indexed from 1 to n, then the following algorithm is optimal because the operation $\leq$ is executed exactly $n-1$ times.

  $m := a[1]$;
  for $i := 2$ to n
  $m :=$ if $m \leq a[i]$ then m else $a[i]$

Example: Simple Sort

- We construct a simple algorithm to sort an array a of numbers indexed from 1 to n as follows:
  - Find the smallest element in a, and exchange it with the first element.
  - Then find the smallest element in positions 2 through n, and exchange it with the element in position 2.
  - Continue in this manner to obtain a sorted array.
- We use a function min and a procedure exchange:
  - $\min(a, i, n)$ is the index of the minimum number among the elements $a[i]$, $a[i+1]$, ..., $a[n]$.
    We can easily modify the algorithm in the previous example to accomplish this task with $n - i$ comparisons.
  - $\text{exchange}(a[i], a[j])$ represents the usual operation of swapping elements and does not use any comparisons.
- We can write the sorting algorithm as follows:

  for $i := 1$ to $n - 1$
      $j := \min(a, i, n)$;
      $\text{exchange}(a[i], a[j])$

Example: Simple Sort (Cont'd)

- We compute the number of comparison operations in simple sort.
- The algorithm for $\min(a, i, n)$ makes $n - i$ comparisons.
- So as i moves from 1 to $n - 1$, the number of comparison operations moves from $n - 1$ to $n - (n - 1) = 1$.
- Adding these comparisons gives the arithmetic expression

$$(n-1) + (n-2) + \cdots + 1 = \frac{n(n-1)}{2}.$$

- We note that there are many faster sorting algorithms.

Analysis of a Loop

- Count the number of times that a procedure S is executed in the following program fragment, where n is a positive integer:

  $i := 1;$
  while $i \leq n$
  $\quad S(i);$
  $\quad i := i + 2$

  Since S occurs in the body of the while loop, we need to count the number of times the body of the loop is executed.

  The body is entered each time i takes on one of the k values $1, 3, 5, \ldots, 2k-1$, where $2k-1 \leq n < 2k+1$.

  With this information, we can express k as a function of n:

$$\begin{array}{rcl} 2k-1 \leq n < 2k+1 & \Rightarrow & 2k \leq n+1 < 2k+2 \\ & \Rightarrow & k \leq \frac{n+1}{2} < k+1. \end{array}$$

  Using the floor function, we conclude $k = \lfloor \frac{n+1}{2} \rfloor$.

Another Analysis of a Loop

- Count the number of times that a procedure S is executed in the following program fragment, where n is a positive integer:

  while $n \geq 1$
  $S(n)$;
  $n := \lfloor n/2 \rfloor$

  S is executed each time the body of the while loop is entered.

  That happens for each of the k values n, $\lfloor \frac{n}{2} \rfloor$, $\lfloor \lfloor \frac{n}{2} \rfloor /2 \rfloor = \lfloor \frac{n}{4} \rfloor$, $\ldots$, $\lfloor \frac{n}{2^{k-1}} \rfloor$, where $\lfloor \frac{n}{2^{k-1}} \rfloor \geq 1 > \lfloor \frac{n}{2^k} \rfloor$.

  Since $\lfloor \frac{n}{2^{k-1}} \rfloor \geq 1$, it follows that $\frac{n}{2^{k-1}} \geq 1$.

  Because $1 > \lfloor \frac{n}{2^k} \rfloor$, we have $\lfloor \frac{n}{2^k} \rfloor = 0$. So $\frac{n}{2^k} < 1$ implying $\frac{n}{2^{k-1}} < 2$.

  Now we have

$$1 \leq \frac{n}{2^{k-1}} < 2 \Rightarrow 2^{k-1} \leq n \leq 2^k \Rightarrow k - 1 \leq \log_2 n < k.$$

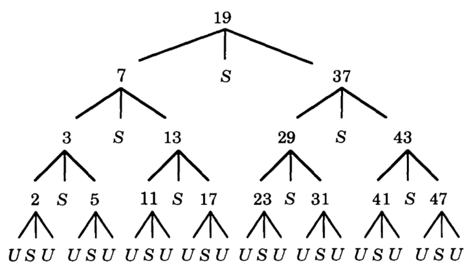
  Using the floor function, we have $k - 1 = \lfloor \log_2 n \rfloor$, i.e., $k = \lfloor \log_2 n \rfloor + 1$.

Binary Search and Decision Trees

- Suppose we search the sorted list below in a binary fashion.

$$2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47.$$

  - We check the middle element of the list to see whether it is the key we are looking for.
  - If not, then we perform the same operation on either the left half or the right half of the list, depending on the value of the key.
- This algorithm has a nice representation as a decision tree:



Binary Search (Cont'd)

- It is easy to see based on the decision tree that there will be at most four comparisons to find whether a number K is in the list.
- So a worst case lower bound for the number of comparisons is 4, which is 1 plus the depth of the binary tree whose nodes are the numbered nodes in the figure.
- The minimum depth of a binary tree with n nodes is $\lfloor \log_2 n \rfloor$.
- So the lower bound for the worst case of a binary search algorithm on a sorted input list of n elements is

$$1 + \lfloor \log_2 n \rfloor.$$

Weighing Things

- Suppose that we are given eight coins that look alike seven of which have identical weight and one is heavier.

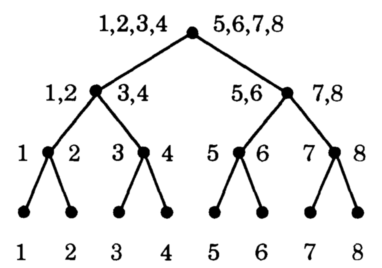
  We want to find the heavy coin among the eight using a pan balance.

  There are two ways to proceed, depending on whether or not we want to consider the possibility that the balance may balance.

  - If the pan never balances, then we will obtain a binary decision tree.
  - Otherwise, we get a ternary decision tree.

Weighing Things: Binary Tree

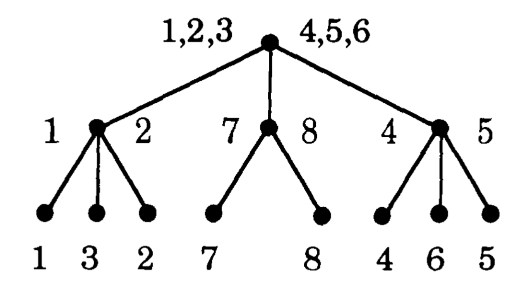
- Let each internal node of the tree represent the pan balance, with an equal number of coins on each side.
- If the left side goes down, then the heavy coin is on the left side.
- Otherwise, the heavy coin is on the right side of the balance.
- Each leaf represents one coin that is the heavy coin.
- One algorithm's decision tree is pictured:



- This algorithm finds the heavy coin in three weighings.

Weighing Things: Ternary Tree

- Since we allow for the third possibility that the two pans are balanced, we do not have to use all eight coins on the first weighing.



- There is no middle branch on the middle subtree, since at this point, one of the coins 7 or 8 must be the heavy one.
- This algorithm finds the heavy coin in two weighings.

Weighing Things: Optimality

- The second solution is an optimal pan balance algorithm for this problem, where we are counting the number of weighings to find the heavy coin.
- Any one of the eight coins could be the heavy one.

  Therefore there must be at least eight leaves on any algorithm's decision tree.

  A ternary tree of depth k can have 3^k possible leaves.

  To get eight leaves, we must have $3^k \geq 8$, or $k \geq 2$.

  Therefore 2 is a lower bound for the number of weighings.
- Since the second solution solves the problem in two weighings, it is optimal.

Subsection 2

Summations and Closed Forms

Sums and Basic Properties of Sums

- The sum of n terms $a_1, a_2, \ldots, a_n$, is denoted

$$\sum_{k=1}^{n} a_k = a_1 + a_2 + \cdots + a_n.$$

- Basic Properties of Sums:
  (a) $\sum_{k=m}^{n} c = (n-m+1)c$;
  (b) $\sum_{k=m}^{n} ca_k = c\sum_{k=m}^{n} a_k$;
  (c) $\sum_{k=1}^{n}(a_k - a_{k-1}) = a_n - a_0$ and $\sum_{k=1}^{n}(a_{k-1} - a_k) = a_0 - a_n$;
  (d) $\sum_{k=m}^{n}(a_k + b_k) = \sum_{k=m}^{n} a_k + \sum_{k=m}^{n} b_k$;
  (e) $\sum_{k=m}^{n} a_k = \sum_{k=m}^{i} a_k + \sum_{k=i+1}^{n} a_k,\ m \leq i < n$;
  (f) $\sum_{k=m}^{n} a_k x^{k+i} = x^i \sum_{k=m}^{n} a_k x^k$;
  (g) $\sum_{k=m}^{n} a_{k+i} = \sum_{k=m+i}^{n+i} a_k$.

Sum of First Powers

- Show that $\sum_{k=1}^{n} k = \frac{n(n+1)}{2}$.
  Note the equation $k^2 - (k-1)^2 = -1 + 2k$.
  Sum from 1 to n to get
  $$\sum_{k=1}^{n}(k^2 - (k-1)^2) = \sum_{k=1}^{n}(-1 + 2k).$$
  This gives
  $$n^2 = \sum_{k=1}^{n}(-1) + \sum_{k=1}^{n} 2k = -n + 2\sum_{k=1}^{n} k.$$
  Solving for $\sum_{k=1}^{n} k$, we obtain
  $$\sum_{k=1}^{n} k = \frac{n^2 + n}{2} = \frac{n(n+1)}{2}.$$

Sums of Squares

- We compute

$$k^3 - (k-1)^3 = k^3 - (k^3 - 3k^2 + 3k - 1) = 1 - 3k + 3k^2.$$

Sum from 1 to n to obtain
$\sum_{k=1}^n (k^3 - (k-1)^3) = \sum_{k=1}^n (1 - 3k + 3k^2)$.
This gives

$$n^3 = \sum_{k=1}^n 1 - 3\sum_{k=1}^n k + 3\sum_{k=1}^n k^2 = n - 3\frac{n(n+1)}{2} + 3\sum_{k=1}^n k^2.$$

Thus, solving for $3\sum_{k=1}^n k^2$, we get

$$\begin{aligned} 3\sum_{k=1}^n k^2 &= n^3 + 3\tfrac{n(n+1)}{2} - n = \tfrac{2n^3+3n(n+1)-2n}{2} \\ &= \tfrac{n(2n^2+3n+1)}{2} = \tfrac{n(n+1)(2n+1)}{2}. \end{aligned}$$

We conclude that $\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}$.

Sum of a Geometric Progression

- Find the sum of the geometric progression $1, a, a^2, \ldots, a^n$, where $a \neq 1$.
  We have $a^{k+1} - a^k = (a-1)a^k$.
  Sum from 0 to n to get
  $$\sum_{k=0}^{n}(a^{k+1} - a^k) = \sum_{k=0}^{n}(a-1)a^k.$$
  Thus, we obtain
  $$a^{n+1} - 1 = (a-1)\sum_{k=0}^{n} a^k.$$
  Therefore,
  $$\sum_{k=0}^{n} a^k = \frac{a^{n+1}-1}{a-1}.$$

A Sum of Products

- Note
$$\begin{aligned}(k+1)a^{k+1}-ka^k &= ka^{k+1}+a^{k+1}-ka^k\\ &= (a-1)ka^k+a^{k+1}.\end{aligned}$$
Sum from $k=1$ to n:
$$\begin{aligned}\sum_{k=1}^n[(k+1)a^{k+1}-ka^k] &= \sum_{k=1}^n[(a-1)ka^k+a^{k+1}]\\ (n+1)a^{n+1}-a &= (a-1)\sum_{k=1}^n ka^k+\sum_{k=1}^n a^{k+1}\\ &= (a-1)\sum_{k=1}^n ka^k+a\sum_{k=1}^n a^k\\ &= (a-1)\sum_{k=1}^n ka^k+a(\sum_{k=0}^n a^k-1)\\ &= (a-1)\sum_{k=1}^n ka^k+a(\frac{a^{n+1}-1}{a-1}-1).\end{aligned}$$
We now get
$$\begin{aligned}(a-1)\sum_{k=1}^n ka^k &= (n+1)a^{n+1}-a-\frac{a^{n+2}-a}{a-1}+a\\ &= \frac{(n+1)a^{n+2}-(n+1)a^{n+1}-a^2+a-a^{n+2}+a+a^2-a}{a-1}\\ &= \frac{a-(n+1)a^{n+1}+na^{n+2}}{a-1}.\end{aligned}$$
We conclude that $\sum_{k=1}^n ka^k=\frac{a-(n+1)a^{n+1}+na^{n+2}}{(a-1)^2}$.

Example: Polynomial Problem

- Compute the number of arithmetic operations needed to evaluate the following polynomial at some number x:

$$c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n.$$

  Suppose that we compute each term in isolation and then add up all the terms.

  There are n addition operations.

  Each term of the form $c_i x^i$ takes i multiplication operations.

  So the total number of arithmetic operations is given by the following sum:

$$\begin{array}{rcl} n + (0 + 1 + 2 + \cdots + n) & = & n + \sum_{k=0}^{n} k \\ & = & n + \frac{n(n+1)}{2} \\ & = & \frac{2n + n^2 + n}{2} \\ & = & \frac{n^2 + 3n}{2}. \end{array}$$

Example: A Simple Sort

- We sort an array a of numbers indexed from 1 to n as follows:
  - Find the smallest element in a and exchange it with the first element.
  - Then find the smallest element in positions 2 through n and exchange it with the element in position 2.
  - Continue in this manner to obtain a sorted array.
- To write the algorithm, we use a function "min" and a procedure "exchange" which are defined as follows:
  - $\min(a, i, n)$ is the index of the minimum number among the elements $a[i], a[i+1], \ldots, a[n]$.
    This task can be accomplished using $n - i$ comparisons.
  - $\text{exchange}(a[i], a[j])$ is the usual operation of swapping elements.
    It does not use any comparisons.
- We write our sorting algorithm:

```
for i := 1 to n − 1
    j := min(a, i, n);
    exchange(a[i], a[j])
```

Example: A Simple Sort (Cont'd)

- Sorting algorithm:

  for $i := 1$ to $n-1$
  $\quad j := \min(a, i, n)$;
  $\quad$ exchange$(a[i], a[j])$

  We compute the number of comparison operations:

  The algorithm for $\min(a, i, n)$ makes $n - i$ comparisons.

  So as i moves from 1 to $n-1$, the number of comparison operations moves from $n-1$ to $n-(n-1) = 1$.

  Adding these comparisons gives the sum of an arithmetic progression,

$$(n-1) + (n-2) + \cdots + 1 = \frac{n(n-1)}{2}.$$

Example: A Problem of Loops

- Assume a procedure $P(j)$ executes $3j$ operations of a certain type. Find the number $T(n)$ of operations executed by P in the following algorithm:

```
i := 1;
while i < n
    i := 2i;
    for j := 1 to i
        P(j)
```

Start by the for-loop: For each i, the for-loop calls $P(1), P(2), P(3), \ldots, P(i)$. Since each call to $P(j)$ executes $3j$ operations, it follows that for each i, the number of operations $f(i)$ executed by the calls on P by the for-loop is

$$f(i) = 3(1+2+3+\cdots+i) = 3\frac{i(i+1)}{2}.$$

Example: A Problem of Loops

- Now we find the values of i that are used to enter the for-loop. The values of i to enter the while-loop are $1, 2, 4, 8, \ldots, 2^k$, where $2^k < n \leq 2^{k+1}$.
  So the values of i to enter the for-loop are $2, 4, 8, \ldots, 2^{k+1}$.
  So we have

$$\begin{array}{rcl} T(n) & = & \sum_{m=1}^{k+1} f(2^m) = \sum_{m=1}^{k+1} \frac{3}{2} 2^m (2^m + 1) \\ & = & \sum_{m=1}^{k+1} 3 \cdot 2^{m-1}(2^m + 1) = 3 \sum_{m=0}^{k} 2^m (2^{m+1} + 1) \\ & = & 3 \sum_{m=0}^{k} (2^{m+1} + 2^m) = 3 \sum_{m=0}^{k} 2^{2m+1} + 3 \sum_{m=0}^{k} 2^m \\ & = & 6 \sum_{m=0}^{k} 4^m + 3 \sum_{m=0}^{k} 2^m = 2(4^{k+1} - 1) + 3(2^{k+1} - 1). \end{array}$$

  Since $2^k < n \leq 2^{k+1}$, we get $k < \log_2 n \leq k + 1$.
  Therefore, $\lceil \log_2 n \rceil = k + 1$.
  Thus, we get

$$\begin{array}{rcl} T(n) & = & 2(4^{\lceil \log_2 n \rceil} - 1) + 3(2^{\lceil \log_2 n \rceil} - 1) \\ & = & 2 \cdot 4^{\lceil \log_2 n \rceil} + 3 \cdot 2^{\lceil \log_2 n \rceil} - 5. \end{array}$$

Sample Approximations

- Suppose we have the following sum of logs.

$$\sum_{k=1}^{n} \log k = \log 1 + \log 2 + \cdots + \log n.$$

- By observing that $\log n$ is the maximum value and $\log 1 = 0$ is the minimum value, we get

$$0 = \log 1 + \log 1 + \cdots + \log 1 \leq \sum_{k=1}^{n} \log k \leq \log n + \log n + \cdots + \log n = n \log n.$$

Sample Approximations (Cont'd)

- We can obtain closer bounds by splitting up the sum and bounding each part:

$$\begin{array}{rcl}\sum_{k=1}^{n}\log k & = & (\log 1+\log 2+\cdots+\log\lfloor\frac{n}{2}\rfloor) \\ & & +(\log(\lfloor\frac{n}{2}\rfloor+1)+\cdots+\log n) \\ & = & \sum_{k=1}^{\lfloor n/2\rfloor}\log k+\sum_{k=\lfloor n/2\rfloor+1}^{n}\log k.\end{array}$$

For a lower bound, we can replace each term of the first sum by 0 and each term of the second sum by $\log(\lfloor\frac{n}{2}\rfloor+1)$.

To get a better upper bound, we can replace each term of the first sum by $\log\lfloor\frac{n}{2}\rfloor$ and each term of the second sum by $\log n$.

This gives us

$$\lceil\frac{n}{2}\rceil\log(\lfloor\frac{n}{2}\rfloor+1)\leq\sum_{k=1}^{n}\log k\leq\lfloor\frac{n}{2}\rfloor(\log\lfloor\frac{n}{2}\rfloor+\log n).$$

Approximating Using Integrals

- We want to approximate a sum of the form

$$\sum_{k=1}^{n} f(k) = f(1) + f(2) + \cdots + f(n),$$

  where f is a continuous function with nonnegative values.
- Each number $f(k)$ can be thought of as the area of a rectangle of with base $[k, k+1]$ and height $f(k)$.
- In this way the sum represents the area of n rectangles, whose bases form the partition $[1,2]$, $[2,3]$, $\ldots$, $[n, n+1]$ of $[1, n+1]$.
- The area under the curve $f(x)$ above the x-axis for x in the closed interval $[1, n+1]$ is given by the definite integral $\int_1^{n+1} f(x)dx$.
- So this definite integral is an approximation for the sum:

$$\sum_{k=1}^{n} f(k) \approx \int_1^{n+1} f(x)dx.$$

Bounds of Monotonic Functions

- Assume f is a monotonic increasing function, which means that $x < y$ implies $f(x) \leq f(y)$.
- Then the area of each rectangle with base $[k, k+1]$ and height $f(k)$ is less than or equal to the area of region under the graph of f on the interval.
- So the integral approximation gives us the following upper bound

$$\sum_{k=1}^{n} f(k) \leq \int_{1}^{n+1} f(x)dx.$$

Bounds of Monotonic Functions (Cont'd)

- For a lower bound notice that the area of each rectangle with base $[k-1, k]$ and height $f(k)$ is greater than or equal to the area of region under the graph of f on the interval.
- So in this case, the sum represents the area of n rectangles, whose bases consist of the partition $[0, 1]$, $[1, 2]$, $\ldots$, $[n-1, n]$ of $[0, n]$.
- So we obtain the following lower bound on the sum

$$\int_0^n f(x)dx \leq \sum_{k=1}^{n} f(k).$$

- So we have

$$\int_0^n f(x)dx \leq \sum_{k=1}^{n} f(k) \leq \int_1^{n+1} f(x)dx.$$

Example

- Find bounds for the following sum, where r is a real number and $r \neq -1$:
$$\sum_{k=1}^{n} k^r = 1^r + 2^r + \cdots + n^r.$$
- If $r = 0$, then the sum becomes $1 + 1 + \ + 1 = n$.
- If $r > 0$, then x^r is increasing for $x \geq 0$.
- So we can obtain bounds using integrals:
$$\sum_{k=1}^{n} k^r \geq \int_0^n x^r dx = \frac{x^{r+1}}{r+1} \Big|_0^n = \frac{n^{r+1}}{r+1};$$
$$\sum_{k=1}^{n} k^r \leq \int_1^{n+1} x^r dx = \frac{x^{r+1}}{r+1} \Big|_1^{n+1} = \frac{(n+1)^{r+1}}{r+1} - \frac{1}{r+1}.$$

So, if $r > 0$, we get
$$\frac{n^{r+1}}{r+1} \leq \sum_{k=1}^{n} k^r \leq \frac{(n+1)^{r+1}}{r+1} - \frac{1}{r+1}.$$

Example (Cont'd)

- If $r < 0$, then x^r is decreasing for $x > 0$, but is not defined at $x = 0$. So the upper bound given in the theory does not exist.

  We work around the problem by raising each lower limit by 1:

$$\int_2^{n+1} x^r dx \leq \sum_{k=2}^{n} k^r \leq \int_1^n x^r dx.$$

  We obtain

$$\begin{array}{c} \sum_{k=2}^n k^r \leq \int_1^n x^r dx = \frac{x^{r+1}}{r+1} \mid_1^n = \frac{n^{r+1}}{r+1} - \frac{1}{r+1}; \\ \sum_{k=2}^n k^r \geq \int_2^{n+1} x^r dx = \frac{x^{r+1}}{r+1} \mid_2^{n+1} = \frac{(n+1)^{r+1}}{r+1} - \frac{2^{r+1}}{r+1}. \end{array}$$

  Adding 1 to each bound gives us the bounds

$$1 + \frac{(n+1)^{r+1}}{r+1} - \frac{2^{r+1}}{r+1} \leq \sum_{k=1}^{n} k^r \leq 1 + \frac{n^{r+1}}{r+1} - \frac{1}{r+1}.$$

Harmonic Numbers

- The *n*-**th harmonic number** is $H_n = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$.
- To approximate H_n, note that:
  - $\frac{1}{x}$ is decreasing for $x > 0$;
  - $\frac{1}{x}$ is undefined for $x = 0$.
- So we use the trick used previously to get

$$\int_2^{n+1} \frac{1}{x} dx \leq \sum_{k=2}^{n} \frac{1}{k} \leq \int_1^n \frac{1}{x} dx.$$

- Using integrals, we find:

$$\ln(n+1) - \ln 2 \leq H_n - 1 \leq \ln n.$$

- Therefore

$$\ln\left(\frac{n+1}{2}\right) + 1 \leq H_n \leq \ln n + 1.$$

A Sum Involving Harmonic Numbers

- Prove the following formula for the sum of the first n harmonic numbers:
$$\sum_{k=1}^{n} H_k = (n+1)H_n - n.$$

$$\begin{array}{rcl}
\sum_{k=1}^{n} H_k & = & H_1 + H_2 + H_3 + \cdots + H_n \\
& = & 1 + (1 + \frac{1}{2}) + (1 + \frac{1}{2} + \frac{1}{3}) + \cdots \\
& & \qquad + (1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}) \\
& = & n \cdot 1 + (n-1)\frac{1}{2} + (n-2)\frac{1}{3} + \cdots + 1 \cdot \frac{1}{n} \\
& = & \sum_{k=1}^{n} (n-k+1)\frac{1}{k} \\
& = & \sum_{k=1}^{n} (n+1)\frac{1}{k} - \sum_{k=1}^{n} k\frac{1}{k} \\
& = & (n+1)\sum_{k=1}^{n} \frac{1}{k} - \sum_{k=1}^{n} 1 \\
& = & (n+1)H_n - n.
\end{array}$$

Sums of Sums

- Assume that a call to $S(j)$ executes about $\frac{n}{j}$ operations.
  Find the number of operations executed by the following algorithm:

  ```
  k := 1;
  while k < n
      k := k + 1;
      for j := 1 to k
          S(j)
  ```

  We examine the for-loop for some value of k: $S(j)$ is called k times with j taking values $1, 2, \ldots, k$. Since $S(j)$ executes $\frac{n}{j}$ operations, the number of operations executed in each for-loop by S is
  $\sum_{j=1}^{k} \frac{n}{j} = n \sum_{j=1}^{k} \frac{1}{j} = nH_k$.
  Now we need to find the values of k at the for-loop: The values of k that enter the while-loop are $1, 2, \ldots, n-1$. Since k gets incremented by 1 upon entry, the values of k at the for-loop are $2, 3, \ldots, n$. So the number of operations by S is given by $\sum_{k=2}^{n} nH_k = n \sum_{k=2}^{n} H_k = n(\sum_{k=1}^{n} H_k - H_1) = n(\sum_{k=1}^{n} H_k - 1) = n(n+1)(H_n - 1)$.

Polynomials: Division

- we consider **rational functions**, i.e., expressions of the form $\frac{p(x)}{q(x)}$, where $p(x)$ and $q(x)$ are two polynomials.
- To work with those fractions we require that the degree of $p(x)$ be less than the degree of $q(x)$.
- If the degree of $p(x)$ is greater than or equal to the degree of $q(x)$, then we can transform the expression into the form

$$\frac{p(x)}{q(x)} = s(x) + \frac{p_1(x)}{q(x)},$$

  where $s(x)$, $p_1(x)$, and $q(x)$ are polynomials and the degree of $p_1(x)$ is less than the degree of $q(x)$.
- The transformation can be carried out by using **long division** for polynomials.

Example

- Perform the long division $(2x^3 + 1) \div (x^2 + 3x + 2)$.

  We have

$$
\begin{array}{r|l}
 & \text{2x-6} \\
\cline{2-2}
x^2 + 3x + 2 & 2x^3 + 1 \\
 & 2x^3 + 6x^2 + 4x \\
\cline{2-2}
 & -6x^2 - 4x + 1 \\
 & -6x^2 - 18x - 12 \\
\cline{2-2}
 & 14x + 13
\end{array}
$$

  So we have

$$\frac{2x^3 + 1}{x^2 + 3x + 2} = 2x - 6 + \frac{14x + 13}{x^2 + 3x + 2}.$$

Polynomials: Partial Fractions

- Assume that we have the quotient $\frac{p(x)}{q(x)}$ of polynomials, where the degree of $p(x)$ is less than the degree of $q(x)$.
- Its **partial fraction** representation or decomposition is a sum of terms that satisfy the following rules, where $q(x)$ has been factored into a product of linear and/or quadratic factors.
  1. If the linear polynomial $ax+b$ is repeated k times as a factor of $q(x)$, then add the following terms to the partial fraction representation,
  $$\frac{A_1}{ax+b}+\frac{A_2}{(ax+b)^2}+\cdots+\frac{A_k}{(ax+b)^k},$$
  where $A_1,\ldots,A_k$ are constants to be determined;
  2. If the quadratic polynomial cx^2+dx+e is repeated k times as a factor of $q(x)$, then add the following terms to the partial fraction representation,
  $$\frac{A_1x+B_1}{cx^2+dx+e}+\frac{A_2x+B_2}{(cx^2+dx+e)^2}+\cdots+\frac{A_kx+B_k}{(cx^2+dx+e)^k},$$
  where A_i and B_i are constants to be determined.

Examples

- $\frac{x-1}{x(x-2)(x+1)} = \frac{A}{x} + \frac{B}{x-2} + \frac{C}{x+1}$;
- $\frac{x^3-1}{x^2(x-2)^3} = \frac{A}{x} + \frac{B}{x^2} + \frac{C}{x-2} + \frac{D}{(x-2)^2} + \frac{E}{(x-2)^3}$;
- $\frac{x^2}{(x-1)(x^2+x+1)} = \frac{A}{x-1} + \frac{Bx+C}{x^2+x+1}$;
- $\frac{x}{(x-1)(x^2+1)^2} = \frac{A}{x-1} + \frac{Bx+C}{x^2+1} + \frac{Dx+E}{(x^2+1)^2}$;

Determining the Constants

- Decompose $\frac{x+1}{(2x-1)(3x-1)}$ into partial fractions.

$$\begin{aligned}
&\frac{x+1}{(2x-1)(3x-1)} = \frac{A}{2x-1} + \frac{B}{3x-1} \\
&\Rightarrow x+1 = A(3x-1) + B(2x-1) \\
&\Rightarrow x+1 = (3A+2B)x + (-A-B) \\
&\Rightarrow \left\{ \begin{array}{rcl} 3A+2B & = & 1 \\ -A-B & = & 1 \end{array} \right\} \Rightarrow \left\{ \begin{array}{rcl} A & = & 3 \\ B & = & -4 \end{array} \right.
\end{aligned}$$

So we have $\frac{x+1}{(2x-1)(3x-1)} = \frac{3}{2x-1} - \frac{4}{3x-1}$.

Example: Partial Fractions and Collapsing Sums

- Consider the summation
$$\sum_{k=1}^{n} \frac{1}{k^2+k}.$$
The summand has a partial fraction representation as
$$\begin{array}{l}\frac{1}{k^2+k} = \frac{1}{k(k+1)} = \frac{A}{k} + \frac{B}{k+1} \\ \Rightarrow 1 = A(k+1) + Bk \Rightarrow 1 = (A+B)k + A \\ \Rightarrow \left\{\begin{array}{rcl} A+B & = & 0 \\ A & = & 1 \end{array}\right\} \Rightarrow \left\{\begin{array}{rcl} A & = & 1 \\ B & = & -1 \end{array}\right.\end{array}$$
So we have $\frac{1}{k^2+k} = \frac{1}{k} - \frac{1}{k+1}$.
So we can compute the sum as follows:
$$\sum_{k=1}^{n} \frac{1}{k^2+k} = \sum_{k=1}^{n} (\frac{1}{k} - \frac{1}{k+1}) = \frac{1}{1} - \frac{1}{n+1} = \frac{n}{n+1}.$$

Example: Sum with a Harmonic Answer

- Evaluate the sum $\displaystyle\sum_{k=1}^{n}\frac{14k+13}{k^2+3k+2}$.

  We can use partial fractions:

$$\begin{aligned}&\tfrac{14k+13}{k^2+3k+2} = \tfrac{14k+13}{(k+1)(k+2)} = \tfrac{A}{k+1} + \tfrac{B}{k+2}\\ &\Rightarrow 14k+13 = A(k+2)+B(k+1)\\ &\Rightarrow 14k+13 = (A+B)k+(2A+B)\\ &\Rightarrow \left\{\begin{array}{rcl} A+B &=& 14\\ 2A+B &=& 13\end{array}\right\} \Rightarrow \left\{\begin{array}{rcl} A &=& -1\\ B &=& 15\end{array}\right.\end{aligned}$$

  So we can calculate the sum as follows:

$$\begin{array}{rcl}\sum_{k=1}^{n}\frac{14k+13}{k^2+3k+2} &=& \sum_{k=1}^{n}(\frac{15}{k+2}-\frac{1}{k+1})\\ &=& 15\sum_{k=1}^{n}\frac{1}{k+2}-\sum_{k=1}^{n}\frac{1}{k+1}\\ &=& 15\sum_{k=3}^{n+2}\frac{1}{k}-\sum_{k=2}^{n+1}\frac{1}{k}\\ &=& 15(H_{n+2}-H_2)-(H_{n+1}-H_1).\end{array}$$

Subsection 3

Permutations and Combinations

The Rule of Sum and the Rule of Product

- **Rule of Sum**:

  If there are m choices for some event to occur and n choices for another event to occur and the events are disjoint, then there are $m + n$ choices for either event to occur.

- This is just another way to say that the cardinality of the union of disjoint sets is the sum of the cardinalities of the two sets.
- **Rule of Product**:

  If there are m choices for some event and n choices for another event, then there are mn choices for both events to occur.

- This is just another way to say that the cardinality of the Cartesian product of two finite sets is the product of the cardinalities of the two sets.

Examples

- Suppose a vending machine has six types of drinks and 18 types of snack food.
  (a) How many choices are there in the machine?
  (b) In how many ways can a customer choose both a drink and a snack?

(a) There are $6 + 18 = 24$ possible choices in the machine.

(b) There are $6 \cdot 18 = 108$ possible choices of a drink and a snack.

- Suppose we have a bookshelf with 5 technical books, 12 biographies, and 37 novels.

  In how many ways can we choose two books of different types?

  There are:
  - $5 \cdot 12 = 60$ ways to choose a technical book and a biography;
  - $5 \cdot 37 = 185$ ways to choose a technical book and a novel;
  - $12 \cdot 37 = 444$ ways to choose a biography and novel.

  So there are $50 + 185 + 444 = 689$ ways to choose two books of different types.

Permutations

- A **permutation** of a set S of n elements is an ordered arrangement of the elements of S.
- There are $n!$ permutations of a set of n elements.
  We reason as follows:
  - There are n choices for the first element.
  - For each of these choices there are $n-1$ choices for the second element.

    $\vdots$
  - Finally, there is only one choice left for the last element.

  We obtain $n \cdot (n-1) \cdot \cdots \cdot 2 \cdot 1 = n!$ different permutations of n elements.
- Example: If $S = \{a, b, c\}$, then there are $3! = 6$ possible permutations of S:

$$abc, \ acb, \ bac, \ bca, \ cab, \ cba.$$

r-Permutations of n Elements

- The number of permutations of r elements chosen from a set of n elements, where $1 \leq r \leq n$, is $\frac{n!}{(n-r)!}$.
  - There are n choices for the first element.
  - For each of these choices there are $n-1$ choices for the second element.

    $\vdots$
  - We continue this process r times, with $n-r+1$ remaining choices for the last element.

  We get $n(n-1)\cdots(n-r+1) = \frac{n(n-1)\cdots(n-r+1)(n-r)\cdots 1}{(n-r)\cdots 1} = \frac{n!}{(n-r)!}$.
- This number is denoted by the symbol $P(n,r)$, read "the number of permutations of n objects taken r at a time".
- So we have

$$P(n,r) = n \cdot (n-1) \cdots (n-r+1) = \frac{n!}{(n-r)!}.$$

Example

- Suppose $S = \{a, b, c, d\}$.
  (a) Find the number of 2-permutations of S.
  (b) List all 2-permutations of S.

(a) The number of 2-permutations of S is

$$P(4,2) = \frac{4!}{(4-2)!} = \frac{4!}{2!} = 12.$$

(b) The 12 permutations are listed as follows:

$$ab,\ ba,\ ac,\ ca,\ ad,\ da,\ bc,\ cb,\ bd,\ db,\ cd,\ dc.$$

Permutations With Replacement

- If we can pick an element more than once, then the objects are said to be selected/permuted **with replacement** or **with repetitions allowed**.
- The number of permutations of r objects from a set of n elements with replacement is n^r.
  - The first element can be selected in n ways.
  - For each choice, the second element can be selected in n ways.

    $\vdots$
  - Finally, the last element can also be selected in n ways.

  Therefore, there are $n \cdot n \cdots n = n^r$ permutations of r out of n objects with replacement.

Example

- Consider the set $S = \{a, b, c\}$.
  (a) Find the the number of 2-permutations with replacement of S.
  (b) List the 2-permutations with replacement of S.

(a) The number of 2-permutations with replacement of S is

$$3^2 = 9.$$

(b) They can be listed as

$$aa, \ ab, \ ac, \ ba, \ bb, \ bc, \ ca, \ cb, \ cc.$$

Permutations With Fixed Number of Repetitions

- Let S be a set with k elements.
- An $(m_1, m_2, \ldots, m_k)$-**permutation** of S is a permutation of $n = m_1 + m_2 + \cdots + m_k$ elements with replacement in which:
    - Element 1 occurs m_1 times;
    - Element 2 occurs m_2 times;

      $\vdots$
    - Element k occurs m_k times.
- The number of $(m_1, m_2, \ldots, m_k)$-permutations of a set of n elements is

$$\frac{n!}{m_1! m_2! \cdots m_k!}.$$

Example

- Let $S = \{a, b\}$.
  (a) Find the number of $(2,3)$-permutations of S.
  (b) List the $(2,3)$-permutations of S.

(a) We have $k = 2,\ n = 5,\ m_1 = 2,\ m_2 = 3$.

So the number $(2,3)$-permutations of S is

$$\frac{5!}{2! \cdot 3!} = 10.$$

(b) The $(2,3)$-permutations of S are:

$$aabbb,\ ababb,\ abbab,\ abbba,\ baabb,$$
$$babab,\ babba,\ bbaab,\ bbaba,\ bbbaa.$$

Example: Worst Case Lower Bound for Comparison Sorting

- Find a lower bound for the number of comparison operations performed by any sorting algorithm that sorts by comparing elements in the list to be sorted.

  Assume that we have a set of n distinct numbers.

  There are $n!$ possible arrangements of these numbers.

  So any decision tree for a comparison sorting algorithm must contain at least $n!$ leaves, one leaf for each possible outcome of sorting one arrangement.

  We know that a binary tree of depth d has at most 2^d leaves.

  So the depth d of the decision tree for any comparison sort of n items must satisfy the inequality $n! \leq 2^d$.

  Solving this inequality for d we get:

$$\log_2(n!) \leq d \Rightarrow \lceil \log_2(n!) \rceil \leq d.$$

Circular Arrangements

- In how many ways can 20 people be arranged in a circle if we do not count a rotation of the circle as a different arrangement?

  There are 20! arrangements of 20 people in a line.

  We can form a circle by joining the two ends of a line.

  However, the same circle is obtained by 20 distinct arrangements of the people.

  It follows that there are $\frac{20!}{20} = 19!$ distinct arrangements of 20 people in a circle.
- Another way to proceed is to fix one person at which the circle will be broken.

  Then we fill in the remaining 19 people in all possible ways to get 19! arrangements.

Example

- How many distinct strings can be made by rearranging the letters of the word

  $$banana$$

  The distinct strings represent the $(3, 1, 2)$-permutations of the set $S = \{a, b, n\}$.

  We have

  $$k = 3, \quad n = 6, \quad m_1 = 3, \quad m_2 = 1, \quad m_3 = 2.$$

  So the number of strings is

  $$\frac{6!}{3!1!2!} = 60.$$

Example

- How many distinct strings of length 10 can be constructed from the digits 0 and 1 with the restriction that five characters must be 0 and five must be 1?

  We are looking for the number of $(5, 5)$-permutations of the set $\{0, 1\}$

  We get
  $$\frac{10!}{5!5!} = 252.$$

Combinations

- An r-**combination** of a set of n elements is a selection of r of the n elements (in which order is not important).
- The number of r-combinations of a set of n elements is $C(n,r) = \frac{n!}{r!(n-r)!}$.

  The key is to notice that to form an r-permutation:
  - First choose an r-combination;
  - For each choice, order the r elements selected.

  Using the Rule of Product, we get the equation: $P(n,r) = C(n,r) \cdot r!$.

  Therefore,

$$C(n,r) = \frac{P(n,r)}{r!} = \frac{\frac{n!}{(n-r)!}}{r!} = \frac{n!}{r!(n-r)!} =: \binom{n}{r}.$$

Example

- Let $S = \{a, b, c, d\}$.
  (a) Find the number of 3-combinations of S.
  (b) List all 3-combinations of S.

(a) We have $n = 4$, $r = 3$.

So we get

$$C(4,3) = \frac{4!}{3!(4-3)!} = 4.$$

(b) The 3-combinations of S are:

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

Algebraic vs. Combinatorial Proofs

- Prove that $\binom{n}{k} = \binom{n}{n-k}$.
  - **Algebraic Proof**: We expand and simplify:

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

  - **Combinatorial Proof**: Reason that both sides are counting the number of ways of performing essentially the same task (following different strategies).
    - The left side is the number of ways to select k objects out of a set of n objects to "take".
    - The right side is the number of ways to select $n-k$ objects out of a set of n objects to "leave out".

  Either selection amounts to the same thing.
  Therefore $\binom{n}{k} = \binom{n}{n-k}$.

An Algebraic Proof

- Prove that $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$.

  We expand and simplify:

$$\begin{array}{rcl}
\binom{n-1}{k} + \binom{n-1}{k-1} & = & \frac{(n-1)!}{k!(n-1-k)!} + \frac{(n-1)!}{(k-1)!(n-1-k+1)!} \\
& = & \frac{(n-1)!}{(k-1)!(n-k-1)!}\left[\frac{1}{k} + \frac{1}{n-k}\right] \\
& = & \frac{(n-1)!}{(k-1)!(n-k-1)!}\frac{n-k+k}{k(n-k)} \\
& = & \frac{(n-1)!n}{(k-1)!k(n-k-1)!(n-k)} \\
& = & \frac{n!}{k!(n-k)!} \\
& = & \binom{n}{k}.
\end{array}$$

A Combinatorial Proof

- Prove that $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$.

  Suppose that we want to count the k-combinations of a set of n elements.

  We may use one of two different ways:

  - The first way to do the selection directly in $\binom{n}{k}$ ways.
  - Alternatively, we may consider a distinguished element x.
    The k-combinations can be partitioned into those that include x and those that do not include x.
    - The number of those that include x is the number of the $(k-1)$-combinations of the remaining $n-1$ elements.
    - The number of those that do not include x is the number of k-combinations of the remaining $n-1$ elements.

    Thus, the total number is $\binom{n-1}{k} + \binom{n-1}{k-1}$.

  Since both ways count the number of k-combinations of a set of n elements, they are equal: $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$.

Combinations With Replacement

- Suppose we have an n element set S.
  The number of k-combinations of S allowing repetitions is $\binom{n+k-1}{n-1}$.
  Imagine that we select:
  - m_1 elements of type 1;
  - m_2 elements of type 2;

    $\vdots$
  - m_n elements of type n.

  Here $m_1, m_2, \ldots, m_n \geq 0$ and $m_1 + \cdots + m_n = k$.
  We think of this selection written as a string where elements of the same type are written together separated by $|$.
  Thus, such a combination amounts to the placement of the $n-1$ separators in the string of $n+k-1$ symbols.
  This placement can be done in as many ways as selecting $n-1$ positions out of a total of $n+k-1$ distinct positions, i.e., in $\binom{n+k-1}{n-1}$ ways.

Example

- In how many ways can 4 coins be selected from a collection of pennies, nickels, and dimes?

  Let $S = \{\text{penny}, \text{nickel}, \text{dime}\}$.

  Then we need the number of 4-combinations of S with replacement.

  The answer is
  $$\binom{3+4-1}{3-1} = \binom{6}{2} = 15.$$

Example

- In how many ways can five people be selected from a collection of Democrats, Republicans and Independents?

  Let
  $$S = \{\text{Democrat}, \text{Republican}, \text{Independent}\}.$$

  We want the number of 5-combinations of S allowing repetitions.

  The answer is
  $$\binom{3+5-1}{3-1} = \binom{7}{2} = 21.$$

Subsection 4

Discrete Probability

Sample Spaces and Probability Distributions

- A **sample space** is the set of all possible **outcomes** of an experiment.
- Outcomes are also called (**sample**) **points**.
- Example: Consider the experiment of tossing a coin twice.
  The sample space for this experiment is $S = \{HH, HT, TH, TT\}$.
- A **probability distribution** on a sample space $S = \{x_1, x_2, \ldots, x_n\}$ is a function $P : S \to [0, 1]$, such that

$$P(x_1) + P(x_2) + \cdots + P(x_n) = 1.$$

- Example (Cont'd): If the coin tossed is fair, we may adopt the following probability distribution on $S = \{HH, HT, TH, TT\}$:

$$P(HH) = P(HT) = P(TH) = P(TT) = \frac{1}{4}.$$

Events and Probabilities

- An **event** E in a sample space S is a subset of S, i.e., a set of some outcomes of the experiment.
- Example (Cont'd): The event E of getting at least one heads in two tosses of a coin is $E = \{HH, HT, TH\}$.
- The **probability** $P(E)$ of an event E in S is the sum of the probabilities of the outcomes included in E:

$$P(E) = \sum_{x \in E} P(x).$$

- Example (Cont'd): Assuming $S = \{HH, HT, TH, TT\}$ and $P(HH) = P(HT) = P(TH) = P(TT) = \frac{1}{4}$, we have

$$P(\{HH, HT, TH\}) = P(HH) + P(HT) + P(TH) = \frac{3}{4}.$$

Elementary Properties

- Let S be a sample space with probability distribution P and $E \subseteq S$.
  (a) $P(\emptyset) = 0$;
  (b) $P(E') = 1 - P(E)$;
  (c) $P(S) = 1$.

(a) $P(\emptyset) = \sum_{x \in \emptyset} P(x) = \sum \emptyset = 0$.

(b) By definition $\sum_{x \in S} P(x) = 1$.
Therefore, $\sum_{x \in E} P(x) + \sum_{x \in E'} P(x) = 1$.
This gives $\sum_{x \in E'} P(x) = 1 - \sum_{x \in E} P(x)$.
The last is rewritten as $P(E') = 1 - P(E)$.

(c) $P(S) = \sum_{x \in S} P(x) = 1$.

Union and Mutually Exclusive Events

- **The Union Formula**

  If A and B are two events, then

  $$P(A \cup B) = P(A) + P(B) - P(A \cap B).$$

- Two events A and B are called **mutually exclusive** if $A \cap B = \emptyset$.
- **The Union Formula for Mutually Exclusive Events**

  If A and B are mutually exclusive events, then

  $$P(A \cup B) = P(A) + P(B).$$

Example

- Consider the experiment of tossing a fair coin 5 times.
  What is the probability of getting at least one head?
  The sample space S consists of $2^5 = 32$ outcomes.
  Let E be the event "at least one head".
  Then, we have

$$\begin{array}{rcl} P(E) & = & 1 - P(E') \\ & = & 1 - P(\{TTTTT\}) \\ & = & 1 - \frac{1}{32} \\ & = & \frac{31}{32}. \end{array}$$

Example

- A bowl contains ten tickets numbered $1, 2, \ldots, 10$.
  Three friends are drawing a ticket in turn, with replacement.
  What is the probability of at least two drawing the same ticket?
  The sample space has $10^3 = 1000$ outcomes.
  Since all outcomes are equally likely, each has probability $\frac{1}{1000}$.
  Let E be the event "at least two tickets same".
  Then E' is the event "all three tickets are different".
  Thus, the number of outcomes in E' is

$$P(10,3) = \frac{10!}{(10-3)!} = \frac{10!}{7!} = 8 \cdot 9 \cdot 10 = 560.$$

  Therefore, we get

$$P(E) = 1 - P(E') = 1 - \frac{560}{1000} = \frac{440}{1000} = 0.44.$$

Odds For and Against an Event

- Let S be a sample space and E an event in S.
- The **odds for** (or **odds in favor of**) E is the ratio $\frac{P(E)}{P(E')}$.
- The **odds against** E is the ratio $\frac{P(E')}{P(E)}$.
- Example: Suppose we roll a pair of fair dice.

  Consider the event $E =$ "the total on the two dice adds up to 5".

  We have $E = \{(1,4),(2,3),(3,2),(4,1)\}$.

  Therefore, $P(E) = \frac{4}{36}$ and $P(E') = \frac{32}{36}$.

  So the odds in favor of the roll coming up 5 are $\frac{4}{32} = \frac{1}{8}$.

  This is often read as "1 to 8".
- Given the odds, we can recover the probabilities:

  If the odds in favor of E are $\frac{x}{y}$, then $P(E) = \frac{x}{x+y}$ and $P(E') = \frac{y}{x+y}$.

Conditional Probability

- If A and B are events in a sample space and $P(B) \neq 0$, then the **conditional probability of A given B** is denoted by $P(A|B)$ and defined by
$$P(A|B) = \frac{P(A \cap B)}{P(B)}.$$
- Example: Consider the experiment of flipping two fair coins.
  (a) Find the probability of the two flips giving a different outcome.
  (b) What is the conditional probability that the two flips give different outcomes given that at least one came up tails?

(a) Let $E =$ "flips different". Then $E = \{(H,T),(T,H)\}$. So $P(E) = \frac{2}{4} = \frac{1}{2}$.

(b) Let $F =$ "at least one tails".
Then $F = \{(H,T),(T,H),(T,T)\}$.
Moreover $E \cap F = \{(H,T),(T,H)\}$.
So we get $P(E|F) = \frac{P(E \cap F)}{P(F)} = \frac{2/4}{3/4} = \frac{2}{3}$.

The Product Rule for the Intersection

- **The Product Rule**

  Given two events A and B in a sample space, with $P(B) \neq 0$,
  $$P(A \cap B) = P(A|B)P(B).$$

- Example: An ad on a web site is read by 20% of the visitors.

  Moreover, if the ad is read, then the probability that the reader buys the product advertised is 0.005.

  What is the probability that a visitor to the website will read the ad and buy the product?

  We set
  $$\begin{array}{rcl} A & = & \text{“read the ad”;} \\ B & = & \text{“buy the product”.} \end{array}$$

  We know that $P(A) = 0.20$ and $P(B|A) = 0.005$.

  We want to find the probability $P(A \cap B)$:
  $$P(A \cap B) = P(A)P(B|A) = (0.20)(0.005) = 0.001.$$

Bayes' Theorem

- **A priori probability** is a probability of an event happening in the future given past experience.
- **A posteriori probability** is the probability of an even having happened given new, after-the-fact, information.
- **Bayes' Formula**

  Suppose a sample space S is partitioned into mutual exclusive events $H_1, \ldots, H_n$. Let E be another event such that $P(E) \neq 0$.
  Then, for all $i = 1, \ldots, n$, we have

$$\begin{aligned} P(H_i|E) &= \frac{P(H_i \cap E)}{P(H_1 \cap E) + \cdots + P(H_n \cap E)} \\ &= \frac{P(H_i)P(E|H_i)}{P(H_1)P(E|H_1) + \cdots + P(H_n)P(E|H_n)}. \end{aligned}$$

- $P(H_i)$ is the a priori probability of H_i.
- $P(H_i|E)$ is the a posteriori probability of H_i given E.

Example

- Suppose a sports team wins 75% of the games it plays in good weather, but only 50% of the games it plays in bad weather.

  The historic weather pattern for September has good weather $\frac{2}{3}$ of the time and bad weather the rest of the time.

  If we read in the paper that the team has won a game on September 12, what is the probability that the weather was bad on that day?

  We set G and B the events of the weather being good and being bad, respectively, and W the event that the team wins.

  The we know $P(G) = \frac{2}{3}, P(B) = \frac{1}{3}, P(W|G) = \frac{3}{4}, P(W|B) = \frac{1}{2}$.

  Therefore, we can compute the a posteriori probability:

$$\begin{aligned} P(B|W) &= \frac{P(B)P(W|B)}{P(G)P(W|G)+P(B)P(W|B)} \\ &= \frac{\frac{1}{3}\cdot\frac{1}{2}}{\frac{2}{3}\cdot\frac{3}{4}+\frac{1}{3}\cdot\frac{1}{2}} = \frac{\frac{1}{6}}{\frac{1}{2}+\frac{1}{6}} = \frac{1}{4}. \end{aligned}$$

Example

- Suppose the input data set for a program may be any of two types. One makes up 60% of the data and the other makes up 40%. Suppose further that inputs from the two types cause warning messages 15% of the time and 20% of the time, respectively. If a random warning message is received, what are the chances that it was caused by an input of the second type?
  Set T_1 and T_2 be the events that the data are of type 1 and type 2, respectively and W be the event that a warning message occurs. Then we know
  $P(T_1) = \frac{3}{5}, P(T_2) = \frac{2}{5}, P(W|T_1) = \frac{3}{20}, P(W|T_2) = \frac{1}{5}$.
  Therefore, we have

$$\begin{aligned} P(T_2|W) &= \frac{P(T_2)P(W|T_2)}{P(T_1)P(W|T_1)+P(T_2)P(W|T_2)} \\ &= \frac{\frac{2}{5}\cdot\frac{1}{5}}{\frac{3}{5}\cdot\frac{3}{20}+\frac{2}{5}\cdot\frac{1}{5}} = \frac{\frac{2}{25}}{\frac{9}{100}+\frac{2}{25}} = \frac{8}{17}. \end{aligned}$$

Independent Events

- Two events A and B are **independent** if
$$P(A \cap B) = P(A)P(B).$$
- Recall by the product rule $P(A \cap B) = P(A)P(B|A) = P(B)P(A|B)$. Therefore, if A and B are independent, we also get
$$P(B|A) = P(B) \quad \text{and} \quad P(A|B) = P(A).$$
- Example: Consider the experiment of flipping two fair coins. Show that the events $A =$"first coin heads" and $B =$"coins come up different" are independent.

  We have
$$\begin{array}{rcl} A & = & \{(H,H),(H,T)\}; \\ B & = & \{(H,T),(T,H)\}; \\ A \cap B & = & \{(H,T)\}. \end{array}$$
  Therefore
$$P(A \cap B) = \frac{1}{4} = \frac{1}{2} \cdot \frac{1}{2} = P(A)P(B).$$

Binomial Distribution

- Suppose an experiment has two outcomes, "success" with probability p and "failure" with probability $1-p$.
- The experiment is repeated n times, each repetition called a **trial**.
- We assume that the trials are independent.
- The probability of getting k successes in n independent trials is
$$b(n,k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad 0 \leq k \leq n.$$
- The set of these probabilities is called the **binomial distribution**.

Example

- A golfer wins 60% of the tournaments that he enters.

  What is the probability that the golfer will win exactly five of the next seven tournaments?

  We ask for the probability of 5 successes in a series of 7 independent trials, with probability of success $p = \frac{3}{5}$.

  So we have

$$b(7,5) = \binom{7}{5}\left(\frac{3}{5}\right)^5\left(\frac{2}{5}\right)^2.$$

Conditional Independence

- Two events A and B are **conditionally independent given the event** C if
$$P(A \cap B|C) = P(A|C)P(B|C).$$
- If $P(A \cap C) \neq 0$ and $P(B \cap C) \neq 0$, then conditional independence of A and B given C may be equivalently expressed by
$$P(A|B \cap C) = P(A|C) \quad \text{or} \quad P(B|A \cap C) = P(B|C).$$

We show the equivalence of the definition with the first condition:

Let $P(A \cap B|C) = P(A|C)P(B|C)$.

$$\begin{aligned} P(A|B \cap C) &= \frac{P(A \cap B \cap C)}{P(B \cap C)} \\ &= \frac{P(A \cap B|C)P(C)}{P(B \cap C)} \\ &= \frac{P(A|C)P(B|C)P(C)}{P(B \cap C)} \\ &= \frac{P(A|C)P(B \cap C)}{P(B \cap C)} \\ &= P(A|C). \end{aligned}$$

Let $P(A|B \cap C) = P(A|C)$.

$$\begin{aligned} P(A \cap B|C) &= \frac{P(A \cap B \cap C)}{P(C)} \\ &= \frac{P(A|B \cap C)P(B \cap C)}{P(C)} \\ &= \frac{P(A|C)P(B \cap C)}{P(C)} \\ &= \frac{P(A|C)P(B|C)P(C)}{P(C)} \\ &= P(A|C)P(B|C). \end{aligned}$$

Example

- Consider the experiment of:
  1. Selecting one die from a pair of two dice with:
     - one die fair;
     - one die loaded, so that it always comes up 3.
  2. Rolling the selected die twice.

  Consider the three events: $A =$ "the first roll is 3", $B =$"the second roll is 3", $C =$"the selected die is fair".

(a) Are the events A and B independent given C?

$$\begin{array}{rcl} P(A|C) & = & \frac{1}{6}; \\ P(B|C) & = & \frac{1}{6}; \\ P(A \cap B|C) & = & \frac{1}{36}. \end{array}$$

So we have $P(A \cap B|C) = P(A|C)P(B|C)$.

Example (Cont'd)

(b) Are the events A and B independent?

We have

$$\begin{array}{rcl} P(A) & = & P(A \cap C) + P(A \cap C') \\ & = & P(A|C)P(C) + P(A|C')P(C') \\ & = & \frac{1}{6} \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} = \frac{1}{12} + \frac{1}{2} = \frac{7}{12}; \\ P(B) & = & P(B \cap C) + P(B \cap C') \\ & = & P(B|C)P(C) + P(B|C')P(C') \\ & = & \frac{1}{6} \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} = \frac{7}{12}; \\ P(A \cap B) & = & P(A \cap B \cap C) + P(A \cap B \cap C') \\ & = & P(A \cap B|C)P(C) + P(A \cap B|C')P(C') \\ & = & \frac{1}{36} \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} = \frac{1}{72} + \frac{36}{72} = \frac{37}{72}. \end{array}$$

Thus, $P(A \cap B) \neq P(A)P(B)$.

Random Variables

- Consider an experiment with sample space $S = \{s_1, \ldots, s_n\}$.
- A **random variable** is a function $X : S \to V$.
- To calculate the probability that the random variable takes a value $x \in V$, we add the probabilities of all outcomes that X maps to x:

$$P(X = x) = \sum_{i : X(s_i) = x} P(s_i).$$

- Example: Consider the experiment of tossing a fair coin three times. Let $X =$"number of heads that occur".
  Then we have

s	$X(s)$
HHH	3
HHT, HTH, THH	2
HTT, THT, TTH	1
TTT	0

  Therefore, e.g., $P(X = 2) = P(\{HHT, HTH, THH\}) = \frac{3}{8}$.

Expectation

- Let $S = \{s_1, s_2, \ldots, s_n\}$ be a sample space.
- Consider a random variable $X : S \to V$.
- The **expected value** or **expectation** of X is

$$E(X) = \sum_{s \in S} X(s)P(s).$$

or $E(X) = X(s_1)P(s_1) + X(s_2)P(s_2) + \cdots + X(s_n)P(s_n)$.

- It turns out that this is the same as:

$$E(X) = \sum_{x \in V} xP(X = x).$$

- Example: In the experiment of tossing three fair coins, with random variable $X =$"number of heads occurring", find $E(X)$.

$$\begin{aligned} E(X) &= 0 \cdot P(X = 0) + 1 \cdot P(X = 1) \\ &\qquad + 2 \cdot P(X = 2) + 3 \cdot P(X = 3) \\ &= 1 \cdot \tfrac{3}{8} + 2 \cdot \tfrac{3}{8} + 3 \cdot \tfrac{1}{8} = \tfrac{12}{8} = \tfrac{3}{2}. \end{aligned}$$

Example

- Consider the experiment of flipping a single coin.
  - If the coin comes up heads, we agree to pay 4 dollars;
  - If it comes up tails, we agree to accept 5 dollars.

  Find the expected winnings if:

  (a) the coin is fair.

  (b) the coin is biased, with $p(H) = \frac{2}{5}$ and $p(T) = \frac{3}{5}$.

  The sample space is $S = \{H, T\}$.

  We construct a random variable $X : S \to \mathbb{R}$ (reflecting the winnings), with $X(H) = -4$ and $X(T) = 5$.

  Then we have the following:

  (a) $E(X) = X(H)P(H) + X(T)P(T) = (-4)\frac{1}{2} + 5\frac{1}{2} = -\frac{4}{2} + \frac{5}{2} = \frac{1}{2}$;

  (b) $E(X) = X(H)P(H) + X(T)P(T) = (-4)\frac{2}{5} + 5\frac{3}{5} = -\frac{8}{5} + \frac{15}{5} = \frac{7}{5}$.

Average Performance of an Algorithm

- Consider an algorithm A solving a certain problem.
- Let $S = \{I_1, I_2, \ldots, I_k\}$ be a sample space consisting of the set of all possible inputs of size n.
- Define a probability distribution $P : S \to [0, 1]$ on S representing our estimate of how likely it is for each of the inputs of size n to occur.
- Let $X : S \to \mathbb{R}$ be a random variable giving the number of operations required by A on each input of size n.
- Based on the preceding, compute the average number of operations to execute A, or **average performance** of A, as a function of input size n by calculating the following expectation:

$$\mathrm{Avg}_A(n) = \sum_{i=1}^{k} X(I_i)P(I_i).$$

Optimality in the Average Case

- Let A an algorithm solving a given problem.
- Specify a sample space S_n of inputs of size n, for all n.
- Specify a probability distribution $P_n : I_n \to [0,1]$, for all n.
- We say that algorithm A is **optimal in the average case** for the problem, if

  $$\mathrm{Avg}_A(n) \leq \mathrm{Avg}_B(n)$$

  for all $n > 0$ and for all algorithms B that solve the same problem.
- Finding lower bounds for the average case is difficult, as or more so than finding lower bounds for the worst case.

Sequential Search

- Consider the problem of searching for an element k in an array L, indexed from 1 to n.
- Consider the following algorithm A for this problem, which returns the index of the rightmost occurrence of k, if k is in L and returns 0 if k is not in L:

  i:= n;
  while $i \geq 1$ and $k \neq L[i]$
      i:= i-1

- We count the average number of comparisons $k \neq L[i]$ performed by the algorithm.
- Let I_i be the input case where the rightmost occurrence of k is at the ith position of L.

  Let I_{n+1} be the case in which k is not in L.

  The sample space is the set $\{I_1, I_2, \ldots, I_{n+1}\}$.

Sequential Search (Cont'd)

- Let $X(I)$ denote the number of comparisons made by the algorithm when the input has the form I.

  Looking at the algorithm, we obtain the following values:

$$\begin{array}{rcl} X(I_i) & = & n-i+1, \text{ for } 1 \leq i \leq n; \\ X(I_{n+1}) & = & n. \end{array}$$

- Suppose we let q be the probability that k is in L.

  Thus $1-q$ is the probability that k is not in L.

  We also assume that whenever k is in L, its position is random.

  This gives us the following probability distribution p over the sample space:

$$\begin{array}{rcl} P(I_i) & = & \frac{q}{n}, \text{ for } 1 \leq i \leq n; \\ P(I_{n+1}) & = & 1-q. \end{array}$$

Sequential Search (Cont'd)

- The expected number of comparisons made by the algorithm for this probability distribution is given by the expected value of X:

$$\begin{array}{rcl}\mathrm{Avg}_A(n) & = & X(I_1)P(I_1)+\cdots+X(I_{n+1})P(I_{n+1}) \\ & = & \frac{q}{n}(n+(n-1)+\cdots+1)+(1-q)n \\ & = & q\frac{n+1}{2}+(1-q)n.\end{array}$$

- If we know that k is in L, then $q=1$.
  So the expectation is $\frac{n+1}{2}$ comparisons.
- If we know that k is not in L, then $q=0$.
  The expectation is n comparisons.
- If k is in L and it occurs at the first position, then the algorithm takes n comparisons.
- It follows that the worst case occurs for the two input cases I_{n+1} and I_1, and we have $W_A(n)=n$.

Functions of Random Variables

- When a random variable X takes the value x_i, then any expression $f(X)$ takes the value $f(x_i)$.

  So, we have
  $$E(f(X)) = \sum_i f(x_i)P(X = x_i).$$

Joint Probability Distribution

- Suppose X and Y are two random variables for an experiment.
- Suppose X takes on values of the form x_i with probability p_i and Y takes on values of the form y_j with probability q_j.
- For each pair (x_i, y_j) we have the **joint probability** $p(x_i, y_j) = P(X = x_i \text{ and } Y = y_j)$.
- With this definition it can be shown using properties of probability that
$$p_i = \sum_j p(x_i, y_j) \quad \text{and} \quad q_j = \sum_i p(x_i, y_j).$$
- If X and Y take values x_i and y_j, then any expression $g(x, y)$ takes the value $g(x_i, y_j)$.
  So, we also have
$$E(g(X, Y)) = \sum_{i,j} g(x_i, y_j) p(x_i, y_j).$$

Linearity of Expectation

(a) If c and d are constants, then $E(cX+d)=cE(X)+d$.

We have

$$\begin{array}{rcl} E(cX+d) & = & \sum_i (cx_i+d)p_i = \sum_i (cx_ip_i+dp_i) \\ & = & c\sum_i x_ip_i + d\sum_i p_i = cE(X)+d\cdot 1 \\ & = & cE(X)+d. \end{array}$$

(b) $E(X+Y)=E(X)+E(Y)$.

We have

$$\begin{array}{rcl} E(X+Y) & = & \sum_{i,j}(x_i+y_j)p(x_i,y_j) \\ & = & \sum_{i,j} x_ip(x_i,y_j) + \sum_{i,j} y_jp(x_i,y_j) \\ & = & \sum_i x_i \sum_j p(x_i,y_j) + \sum_j y_j \sum_i p(x_i,y_j) \\ & = & \sum_i x_ip_i + \sum_j y_jp_j = E(X)+E(Y). \end{array}$$

Example

- A fair die is rolled twice.

  Find the expected value of the sum of the two numbers occurring.

  Suppose $X =$"outcome of a single roll".

  Then

  $$E(X) = \sum_{i=1}^{6} iP(X = i) = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + \cdots + 6 \cdot \frac{1}{6} = 21 \cdot \frac{1}{6} = \frac{7}{2}.$$

  Thus, we get

  $$E(X + X) = E(X) + E(X) = \frac{7}{2} + \frac{7}{2} = 7.$$

Example

- A fair die is rolled twice in a game of chance.
  - For the first roll the payout is \$24 if the number of dots on top is 3 or 5. Otherwise, the loss is \$6.
  - For the second roll the payout is \$30 if the number of dots on top is 6. Otherwise the loss is \$12.

  What is the expected value of winnings in this game?

  For the first roll let X take on the values 24, -6 with probabilities $\frac{1}{3}$, $\frac{2}{3}$.

  Then $E(X) = 24 \cdot \frac{1}{3} - 6 \cdot \frac{2}{3} = 4$.

  For the second roll let Y take on the values $30, -12$ with probabilities $\frac{1}{6}, \frac{5}{6}$.

  Then $E(Y) = 30 \cdot \frac{1}{6} - 12 \cdot \frac{5}{6} = -5$.

  So, the expected value of the game is

$$E(X + Y) = E(X) + E(Y) = 4 + (-5) = -1.$$

Variance

- Let X be a random variable with possible values $x_1, \ldots, x_n$ and probabilities $p_1, \ldots, p_n$.
- The **variance** of X, which we denote by $\mathrm{Var}(X)$, is defined as follows, where $\mu = E(X)$:

$$\mathrm{Var}(X) = (x_1 - \mu)^2 p_1 + \cdots + (x_n - \mu)^2 p_n = \sum_{i=1}^{n} (x_i - \mu)^2 p_i.$$

- Example: Consider again the coin-flip experiment where the random variable X is the number of heads that occur when a fair coin is flipped three times.
  X takes on the values 0, 1, 2, 3, with probabilities $\frac{1}{8}$, $\frac{3}{8}$, $\frac{3}{8}$, $\frac{1}{8}$.
  We found that $\mu = E(X) = \frac{3}{2}$.
  Now we compute the variance:

$$\begin{array}{rcl}\mathrm{Var}(X) & = & (0 - \frac{3}{2})^2 \cdot \frac{1}{8} + (1 - \frac{3}{2})^2 \cdot \frac{3}{8} + (2 - \frac{3}{2})^2 \cdot \frac{3}{8} + (3 - \frac{3}{2})^2 \cdot \frac{1}{8} \\ & = & \frac{9}{32} + \frac{3}{32} + \frac{3}{32} + \frac{9}{32} = \frac{27}{32}.\end{array}$$

Properties of Variance

(a) If c is a constant, then

$$\mathrm{Var}(c) = 0, \quad \mathrm{Var}(cX) = c^2 \mathrm{Var}(X), \quad \mathrm{Var}(X + c) = \mathrm{Var}(X).$$

(b) $\mathrm{Var}(X) = E(X^2) - E(X)^2$.

We show (b):

$$\begin{array}{rcl}
\mathrm{Var}(X) & = & \sum_{i=1}^{n} (x_i - \mu)^2 p_i \\
& = & \sum_{i=1}^{n} (x_i^2 p_i - 2x_i \mu p_i + \mu^2 p_i) \\
& = & \sum_{i=1}^{n} x_i^2 p_i - 2\mu \sum_{i=1}^{n} x_i p_i + \mu^2 \sum_{i=1}^{n} p_i \\
& = & E(X^2) - 2\mu^2 + \mu^2(1) \\
& = & E(X^2) - \mu^2 = E(X^2) - E(X)^2.
\end{array}$$

Example: Mean and Variance

- Consider the experiment of rolling a fair die.
  Let X be the value that occurs.
  Compute $\mathrm{Var}(X)$.
  First compute
  $$\mu = E(X) = \frac{1}{6}(1+2+3+4+5+6) = \frac{7}{2}.$$
  Now we comnpute
  $$E(X^2) = \frac{1}{6}(1^2+2^2+3^2+4^2+5^2+6^2) = \frac{91}{6}.$$
  Finally, we get
  $$\mathrm{Var}(X) = E(X^2) - E(X)^2 = \frac{91}{6} - \frac{49}{4} = \frac{35}{12}.$$

Standard Deviation and Properties

- The **standard deviation** of X is the square root of variance and is denoted by $\mathsf{SD}(X)$ and also by the Greek letter "sigma", σ:

$$\sigma = \mathsf{SD}(X) = \mathsf{Var}(X).$$

- Since σ denotes the standard deviation, we denote the variance by $\sigma^2 = \mathsf{Var}(X)$.
- The following properties of standard deviation follow directly from the fact that it is the square root of variance:

  If c is a constant, then

$$\sigma(c) = 0, \quad \sigma(cX) = |c|\sigma(X), \quad \sigma(X + c) = \sigma(X).$$

Subsection 5

Recurrences

Recurrence Relations

- Any recursively defined function f with domain $\mathbb{N}$ that computes numbers is called a **recurrence** or a **recurrence relation**.
- When working with recurrences, we often write f_n in place of $f(n)$.
- Example: The following definition is a recurrence:

$$\begin{aligned} f_0 &= 1 \\ f_n &= 2f_{n-1} + n. \end{aligned}$$

- **Solving** a recurrence f means finding an expression for the general term f_n that is not recursive.

Solving by Substitution

- Solve the recurrence $r_0 = 1$, $r_n = 2r_{n-1} + n$ by substitution.
  We work as follows:

$$\begin{array}{rcl} r_n & = & 2r_{n-1} + n \\ & = & 2(2r_{n-2} + (n-1)) + n = 2^2 r_{n-2} + 2(n-1) + n \\ & = & 2^3 r_{n-3} + 2^2(n-2) + 2(n-1) + n \\ & = & 2^4 r_{n-4} + 2^3(n-3) + 2^2(n-2) + 2(n-1) + n \\ & \vdots & \\ & = & 2^n r_0 + 2^{n-1}1 + 2^{n-2}2 + \cdots + 2^2(n-2) + 2^1(n-1) + 2^0 n \\ & = & 2^n + \sum_{i=0}^{n-1} 2^i(n-i) \\ & = & 2^n + n\sum_{i=0}^{n-1} 2^i - \sum_{i=0}^{n-1} i2^i \\ & = & 2^n + n(2^n - 1) - (2 - n2^n + (n-1)2^{n+1}) \\ & = & 2^n(1 + n + n - 2n + 2) - n - 2 \\ & = & 3 \cdot 2^n - n - 2. \end{array}$$

  So $r_n = 3 \cdot 2^n - n - 2$, for all $n \geq 0$.

Solving by Cancelation

- Solve the recurrence

$$r_0 = 1, \quad r_n = 2r_{n-1} + n.$$

We have the following:

$$\begin{array}{rcl} r_n & = & 2r_{n-1} + n \\ 2r_{n-1} & = & 2^2 r_{n-2} + 2(n-1) \\ 2^2 r_{n-2} & = & 2^3 r_{n-3} + 2^2(n-2) \\ & \vdots & \\ 2^{n-1} r_1 & = & 2^n r_0 + 2^{n-1} \cdot 1 \end{array}$$

Adding side by side we get

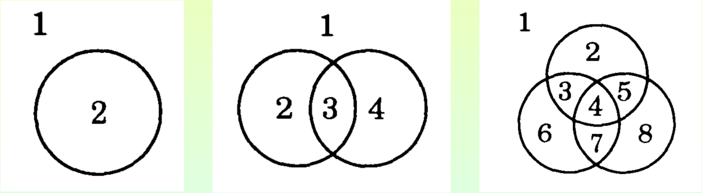
$$r_n = n + 2(n-1) + 2^2(n-2) + \cdots + 2^{n-1} \cdot 1 + 2^n r_0.$$

The n Ovals Problem

- **The n Ovals Problem**

  Suppose that n ovals are drawn on the plane such that no three ovals meet in a point and each pair of ovals intersects in exactly two points. How many distinct regions of the plane are created by n ovals?

- Let r_n be the number of regions created by n ovals.
- We have



$r_1 = 2, \ r_2 = 4, \ r_3 = 8.$

The General Recurrence Relation

- To find a relation involving r_n, consider the following description:
  - $n-1$ ovals divide the plane into r_{n-1} regions.
  - The n-th oval will meet each of the previous $n-1$ ovals in $2(n-1)$ points.
    So the n-th oval will itself be divided into $2(n-1)$ arcs.
  - Each of these $2(n-1)$ arcs splits some region in two.
    Therefore we add $2(n-1)$ regions to r_{n-1} to obtain r_n.
- This gives us the following recursive definition for r_n, which is called a **recurrence**:

$$r_n = r_{n-1} + 2(n-1).$$

Solving Using Substitution

- Consider the recurrence $r_1 = 2$, $r_n = r_{n-1} + 2(n-1)$.

  We solve the recurrence using **substitution**.

$$\begin{aligned} r_n &= r_{n-1} + 2(n-1) \\ &= r_{n-2} + 2(n-2) + 2(n-1) \\ &= r_{n-3} + 2(n-3) + 2(n-2) + 2(n-1) \\ &\vdots \\ &= r_1 + 2 \cdot 1 + 2 \cdot 2 + \cdots + 2(n-3) + 2(n-2) + 2(n-1) \\ &= 2 + 2(1 + 2 + \cdots + (n-1)) \\ &= 2 + 2\frac{n(n-1)}{2} \\ &= n^2 - n + 2. \end{aligned}$$

Solving Using Cancelation

- Consider the recurrence $r_1 = 2$, $r_n = r_{n-1} + 2(n-1)$.

  Solve the recurrence using **cancelation**.

  We have

$$\begin{array}{rcl} r_n & = & r_{n-1} + 2(n-1) \\ r_{n-1} & = & r_{n-2} + 2(n-2) \\ r_{n-2} & = & r_{n-3} + 2(n-3) \\ & \vdots & \\ r_2 & = & r_1 + 2 \cdot 1 \end{array}$$

  So

$$r_n = r_1 + 2(1 + 2 + \cdots + n - 1) = 2 + 2\frac{n(n-1)}{2} = n^2 - n + 2.$$

Polynomial Multiplication

- Suppose that a program computes the value of the polynomial $c_0 + c_1x + c_2x^2 + \cdots + c_nx^n$ by following the procedure

$$c_0 + c_1x + c_2x^2 + \cdots + c_nx^n = c_0 + x(c_1 + c_2x + \cdots + c_nx^{n-1}).$$

Find a recurrence relation for the number T_n of arithmetic operations that are involved in such a computation and solve it to find T_n.
First note that $T_0 = 0$ and $T_1 = 2$.
To compute the value of a polynomial of degree n:
  - We first compute the value of a polynomial of degree $n-1$;
  - Then we perform one multiplication and one addition.

We conclude that the required recurrence is

$$T_0 = 0, \quad T_n = T_{n-1} + 2.$$

Now we have

$$\begin{aligned} T_n &= T_{n-1} + 2 = T_{n-2} + 2 + 2 \\ &= T_{n-3} + 2 + 2 + 2 = \cdots = T_0 + 2n = 2n. \end{aligned}$$

A Divide-and-Conquer Recurrence

- Solve the recurrence $T(1) = 1$, $T(n) = 3T(\frac{n}{2}) + n$, where $n = 2^k$.

  We have

$$\begin{array}{rcl} T(n) & = & 3T(\frac{n}{2}) + n \\ & = & 3^2 T(\frac{n}{4}) + 3\frac{n}{2} + n \\ & = & 3^3 T(\frac{n}{8}) + 3^2 \frac{n}{4} + 3\frac{n}{2} + n \\ & = & \cdots \\ & = & 3^k T(1) + 3^{k-1} \frac{n}{2^{k-1}} + \cdots + 3\frac{n}{2} + n \\ & = & 3^k + n((\frac{3}{2})^{k-1} + \cdots + \frac{3}{2} + 1) \\ & = & 3^k + n\frac{(\frac{3}{2})^k - 1}{\frac{3}{2} - 1} \\ & = & 3^k + 2n[(\frac{3}{2})^k - 1]. \end{array}$$

  For $k = \log_2 n$, we get

$$\begin{array}{rcl} T(n) & = & 3^{\log_2 n} + 2n[(\frac{3}{2})^{\log_2 n} - 1] = 3^{\log_2 n} + 2n[\frac{3^{\log_2 n}}{n} - 1] \\ & = & n^{\log_2 3} + 2n^{\log_2 3} - 2n = 3n^{\log_2 3} - 2n. \end{array}$$

Generating Functions

- Let $a_0, a_1, a_2, \ldots, a_n, \ldots$ be an infinite sequence.
- The **generating function** for this sequence is the expression

$$A(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n + \cdots = \sum_{n=0}^{\infty} a_n x^n.$$

- Expressions of this form are also called **formal power series** or **infinite polynomials**.
- **Equality** of formal power series is defined by

$$a_0 + a_1 x + a_2 x^2 + \cdots = b_0 + b_1 x + b_2 x^2 + \cdots$$

  if and only if $a_0 = b_0$, $a_1 = b_1$, $a_2 = b_2$, $\ldots$,

  i.e., two formal power series are equal if and only if their corresponding coefficients are equal.

Operations on Generating Functions

- The **sum** of two power series is defined by

$$\begin{aligned}&(a_0 + a_1x + a_2x^2 + \cdots) + (b_0 + b_1x + b_2x^2 + \cdots)\\ &\qquad = (a_0 + b_0) + (a_1 + b_1)x + (a_2 + b_2)x^2 + \cdots\end{aligned}$$

- The **product** of two power series is defined by

$$\begin{aligned}&(a_0 + a_1x + a_2x^2 + \cdots)(b_0 + b_1x + b_2x^2 + \cdots)\\ &\qquad = a_0b_0 + (a_0b_1 + a_1b_0)x + (a_0b_2 + a_1b_1 + a_2b_0)x^2 + \cdots\end{aligned}$$

Example: The Geometric Series

- Consider the generating function for the infinite sequence $1, 1, \ldots, 1$.
- This is the **geometric series**:

$$1 + x + x^2 + x^3 + \cdots = \sum_{n=0}^{\infty} x^n.$$

- Note that

$$\begin{aligned}(1-x)&(1+x+x^2+x^3+\cdots)\\ &= 1 + (1-1)x + (1-1)x^2 + (1-1)x^3 + \cdots\\ &= 1.\end{aligned}$$

- Therefore, its closed form is

$$\sum_{n=0}^{\infty} x^n = \frac{1}{1-x}.$$

Example

(a) Find the generating function for the sequence

$$-2, 4, -8, 16, -32, \ldots, (-2)^{n+1}, \ldots$$

We work as follows:

$$\begin{aligned} & -2 + 4x - 8x^2 + 16x^3 - 32x^4 + \cdots \\ = & -2(1 - 2x + 4x^2 - 8x^3 + \cdots) \\ = & -2\sum_{n=0}^{\infty}(-2x)^n = -2\frac{1}{1-(-2x)} = \frac{-2}{1+2x}. \end{aligned}$$

(b) Find the sequence a_n whose generating function is $A(x) = \frac{5}{3-9x}$.
We have

$$\begin{aligned} A(x) &= \frac{5}{3-9x} = \frac{5}{3}\frac{1}{1-3x} \\ &= \frac{5}{3}\sum_{n=0}^{\infty}(3x)^n \\ &= \frac{5}{3}[1 + 3x + 9x^2 + 27x^3 + \cdots + 3^n x^n + \cdots] \end{aligned}$$

Therefore, $a_n = \frac{5}{3}3^n = 5 \cdot 3^{n-1}$.

Example: Partial Fraction Decomposition

- Find the sequence a_n whose generating function is $A(x) = \frac{-x+2}{2x^2+x-1}$.
  First factor the denominator: $A(x) = \frac{-x+2}{(2x-1)(x+1)}$.
  Next decompose into partial fractions:
  $$\begin{array}{rcl} \frac{-x+2}{(2x-1)(x+1)} & = & \frac{A}{2x-1} + \frac{B}{x+1} \\ -x+2 & = & A(x+1) + B(2x-1) \\ -x+2 & = & (A+2B)x + (A-B) \end{array}$$
  So we get $\left\{ \begin{array}{rcl} -1 & = & A+2B \\ 2 & = & A-B \end{array} \right\} \Rightarrow \left\{ \begin{array}{rcl} A & = & 1 \\ B & = & -1 \end{array} \right.$
  Now we have
  $$\begin{array}{rcl} A(x) & = & \frac{-x+2}{(2x-1)(x+1)} = \frac{1}{2x-1} - \frac{1}{x+1} = -\frac{1}{1-2x} - \frac{1}{1-(-x)} \\ & = & -\sum_{n=0}^{\infty}(2x)^n - \sum_{n=0}^{\infty}(-x)^n \\ & = & -\sum_{n=0}^{\infty} 2^n x^n - \sum_{n=0}^{\infty}(-1)^n x^n \\ & = & \sum_{n=0}^{\infty}(-2^n - (-1)^n)x^n. \end{array}$$
  So we get $a_n = -2^n - (-1)^n$.

Solving Recurrences: The Generating Function Method

- Solve the following recurrence relation:

$$a_n = 5a_{n-1} - 6a_{n-2},\ n \geq 2,\ a_0 = 0,\ a_1 = 1.$$

Suppose that $A(x) = a_0 + a_1 x + a_2 x^2 + \cdots$.

1. Obtain an equation involving $A(x)$ as follows:

$$\begin{array}{rcl} \sum_{n=2}^{\infty} a_n x^n & = & \sum_{n=2}^{\infty} (5a_{n-1} - 6a_{n-2}) x^n \\ & = & \sum_{n=0}^{\infty} 5a_{n-1} x^n - \sum_{n=2}^{\infty} 6a_{n-2} x^n \\ & = & 5 \sum_{n=2}^{\infty} a_{n-1} x^n - 6 \sum_{n=2}^{\infty} a_{n-2} x^n. \end{array}$$

Note the following:

$$\begin{array}{rcl} \sum_{n=2}^{\infty} a_n x^n & = & A(x) - a_0 - a_1 x = A(x) - x; \\ \sum_{n=2}^{\infty} a_{n-1} x^n & = & \sum_{n=1}^{\infty} a_n x^{n+1} = x \sum_{n=1}^{\infty} a_n x^n \\ & = & x(A(x) - a_0) = xA(x); \\ \sum_{n=2}^{\infty} a_{n-2} x^n & = & \sum_{n=0}^{\infty} a_n x^{n+2} = x^2 \sum_{n=0}^{\infty} a_n x^n = x^2 A(x). \end{array}$$

So we get $A(x) - x = 5xA(x) - 6x^2 A(x)$.

The Generating Function Method: Step 2

- We found that $A(x) - x = 5xA(x) - 6x^2A(x)$.
  Now we solve for $A(x)$ and attempt to write the solution as an infinite power series:

$$\begin{array}{rcll} A(x) & = & \frac{x}{6x^2-5x+1} & \\ & = & \frac{x}{(2x-1)(3x-1)} & \text{(factor)} \\ & = & \frac{1}{2x-1} - \frac{1}{3x-1} & \text{(partial fractions)} \\ & = & -\frac{1}{1-2x} + \frac{1}{1-3x} & \text{(prepare to expand)} \\ & = & -\sum_{n=0}^{\infty}(2x)^n + \sum_{n=0}^{\infty}(3x)^n & \text{(expand)} \\ & = & -\sum_{n=0}^{\infty} 2^n x^n + \sum_{n=0}^{\infty} 3^n x^n & \\ & = & \sum_{n=0}^{\infty}(-2^n + 3^n)x^n. & \end{array}$$

  Since $A(x) = \sum_{n=0}^{\infty} a_n x^n$, we conclude that

$$a_n = -2^n + 3^n, \quad n \geq 0.$$

Two More Generating Functions

- We have
  (a) $\frac{1}{(1-x)^{k+1}} = \sum_{n=0}^{\infty} \binom{k+n}{n} x^n$, $k \in \mathbb{N}$;
  (b) $(1+x)^r = \sum_{n=0}^{\infty} \frac{r(r-1)\cdots(r-n+1)}{n!} x^n$, $r \in \mathbb{R}$.

(a) This can be shown using a combinatorial argument:

$$\begin{array}{rcl} \frac{1}{(1-x)^{k+1}} & = & (\frac{1}{1-x})^{k+1} \\ & = & (1+x+x^2+\cdots)^{k+1} \\ & & \text{(choose } n \text{ out of } k+1 \text{ objects with replacement)} \\ & = & \sum_{n=0}^{\infty} \binom{n+k}{k} x^n \\ & = & \sum_{n=0}^{\infty} \binom{n+k}{n} x^n. \end{array}$$

(b) This part can be proved using MacLaurin series
$(f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(0)}{n!} x^n)$ for the function $(1+x)^r$.

Example: Parenthesizing a Sum

- Find the number of ways to parenthesize the expression $t_1 + t_2 + \cdots + t_{n-1} + t_n$.
- Example: Some of those ways for the expression $t_1 + t_2 + t_3 + t_4$ are:

$$((t_1 + t_2) + (t_3 + t_4)),\ (t_1 + (t_2 + (t_3 + t_4))),\ (t_1 + ((t_2 + t_3) + t_4)).$$

- To solve the problem, let b_n denote the total number of possible parenthesizations for an n-term expression.
- If $1 \leq k \leq n-1$, then we can split the expression into two subexpressions as follows: $t_1 + \cdots + t_{n-k}$ and $t_{n-k+1} + \cdots + t_n$.
- So there are $b_{n-k} b_k$ ways to parenthesize the entire expression if the final $+$ is placed between the two subexpressions above.
- Letting k range from 1 to $k-1$, we obtain the following formula for b_n when $n \geq 2$:

$$b_n = b_{n-1} b_1 + b_{n-2} b_2 + \cdots + b_2 b_{n-2} + b_1 b_{n-1}.$$

Example: Parenthesizing a Sum (Cont'd)

- If we let $b_0 = 0$ and $b_1 = 1$, the resulting recurrence is

$$b_0 = 0, b_1 = 1, b_n = b_n b_0 + b_{n-1} b_1 + \cdots + b_1 b_{n-1} + b_0 b_n, \ \ n \geq 2.$$

- We solve this recurrence using generating functions:

$$\begin{array}{l} \sum_{n=2}^{\infty} b_n x^n = \sum_{n=2}^{\infty} (b_n b_0 + b_{n-1} b_1 + \cdots + b_1 b_{n-1} + b_0 b_n) x^n \\ B(x) - b_0 - b_1 x = B(x)B(x) - b_0 b_0 - (b_0 b_1 + b_1 b_0) x \\ B(x) - x = B(x)^2 \\ B(x)^2 - B(x) + x = 0 \end{array}$$

- Applying the quadratic formula, we get

$$B(x) = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} = \frac{1 \pm \sqrt{1 - 4x}}{2}.$$

Example: Parenthesizing a Sum (Cont'd)

- We found $B(x) = \frac{1 \pm \sqrt{1-4x}}{2}$.
- We expand $\sqrt{1-4x} = (1-4x)^{1/2}$.

$$\begin{array}{rcl}
(1-4x)^{1/2} & = & (1+(-4x))^{1/2} \\
& = & \sum_{n=0}^{\infty} \frac{\frac{1}{2}(\frac{1}{2}-1)(\frac{1}{2}-2)\cdots(\frac{1}{2}-n+1)}{n!}(-4x)^n \\
& = & 1+\sum_{n=1}^{\infty} \frac{\frac{1}{2}(-\frac{1}{2})(-\frac{3}{2})\cdots(-\frac{2n-3}{2})}{n!}(-2)^n 2^n x^n \\
& = & 1+\sum_{n=1}^{\infty} \frac{(-1)(1)(3)\cdots(2n-3)}{n!} 2^n x^n \\
& = & 1+\sum_{n=1}^{\infty} \frac{(-1)1\cdot 2\cdot 3\cdot 4\cdots(2n-3)(2n-2)}{n!2\cdot 4\cdots(2n-2)} 2^n x^n \\
& = & 1+\sum_{n=1}^{\infty} \frac{(-1)(2n-2)!}{n(n-1)!2^{n-1}1\cdot 2\cdot 3\cdots(n-1)} 2^n x^n \\
& = & 1+\sum_{n=1}^{\infty} \frac{(-1)(2n-2)!}{n(n-1)!(n-1)!} 2x^n \\
& = & 1+\sum_{n=1}^{\infty} (-\frac{2}{n})\binom{2n-2}{n-1} x^n.
\end{array}$$

Example: Parenthesizing a Sum (Cont'd)

- Now we have:

$$B(x) = \frac{1 \pm \sqrt{1-4x}}{2}$$

and

$$(1-4x)^{1/2} = 1 + \sum_{n=1}^{\infty} \left(-\frac{2}{n}\right) \binom{2n-2}{n-1} x^n.$$

- So we get that

$$\begin{array}{rcl} B(x) & = & \frac{1}{2} - \frac{1}{2}\sqrt{1-4x} \\ & = & \frac{1}{2} - \frac{1}{2}\left[1 + \sum_{n=1}^{\infty}(-\frac{2}{n})\binom{2n-2}{n-1}x^n\right] \\ & = & \sum_{n=1}^{\infty} \frac{1}{n}\binom{2n-2}{n-1}x^n. \end{array}$$

- We conclude that

$$b_n = \begin{cases} 0, & \text{if } n = 0 \\ \frac{1}{n}\binom{2n-2}{n-1}, & \text{if } n > 0 \end{cases}$$

Subsection 6

Rates of Growth

Big Oh

- We say **the growth rate of a function** f **is bounded above by the growth rate of a function** g if there are positive numbers c and m such that
$$|f(n)| \leq c|g(n)|, \text{ for } n \geq m.$$
- In this case we write $f(n) = O(g(n))$ and we say that $f(n)$ **is big oh of** $g(n)$.
- Example: If $0 \leq f(n) \leq g(n)$ for all $n \geq m$ for some m, then $f(n) = O(g(n))$ because we can let $c = 1$.
- Example: Suppose that $f_1(n) = O(g(n))$ and $f_2(n) = O(g(n))$. Then there are constants such that $|f_1(n)| \leq c_1|g(n)|$ for $n \geq m_1$, and $|f_2(n)| \leq c_2|g(n)|$ for $n \geq m_2$.
  It follows that, for all $n \geq \max\{m_1, m_2\}$,
$$|f_1(n) + f_2(n)| \leq |f_1(n)| + |f_2(n)| \leq (c_1 + c_2)|g(n)|.$$
  Therefore, $f_1(n) + f_2(n) = O(g(n))$.

Properties of Big Oh

(a) $f(n) = O(f(n))$.

(b) If $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $f(n) = O(h(n))$.

(c) If $0 \leq f(n) \leq g(n)$ for all $n \geq m$, then $f(n) = O(g(n))$.

(d) If $f(n) = O(g(n))$ and a is any real number, then $af(n) = O(g(n))$.

(e) If $f_1(n) = O(g(n))$ and $f_2(n) = O(g(n))$, then

$$f_1(n) + f_2(n) = O(g(n)).$$

(f) If f_1 and f_2 have nonnegative values and $f_1(n) = O(g_1(n))$ and $f_2(n) = O(g_2(n))$, then

$$f_1(n) + f_2(n) = O(g_1(n) + g_2(n)).$$

Polynomials and Big Oh

- If $p(n)$ is a polynomial of degree m or less, then $p(n) = O(n^m)$.

  Let $p(n) = a_0 + a_1 n + \cdots + a_m n^m$.

  If k is an integer such that $0 \leq k \leq m$, then $n^k \leq n^m$ for $n \geq 1$.

  So by Property (c), we have $n^k = O(n^m)$.

  Now we Property (d) gives $a_k n^k = O(n^m)$.

  Finally, Property (e) applied repeatedly to the terms of $p(n)$ yields

  $$p(n) = a_0 + a_1 n + \cdots + a_m n^m = O(n^m).$$

- Example: Suppose we have an algorithm to solve a problem P, whose worst-case running time is a polynomial of degree m.

  Then we can say that an optimal algorithm in the worst case for P, if one exists, must have a worst-case running time of $O(n^m)$.

Big Omega

- We say **the growth rate of f is bounded below by the growth rate of g** if there are positive numbers c and m such that
$$|f(n)| \geq c|g(n)| \text{ for } n \geq m.$$
- In this case we write $f(n) = \Omega(g(n))$ and we say that $f(n)$ **is big omega of** $g(n)$.
- The following relationship holds between big omega and big oh:
$$f(n) = \Omega(g(n)) \quad \text{if and only if} \quad g(n) = O(f(n)).$$
Using the constant c for one of the definitions corresponds to using the constant $\frac{1}{c}$ for the other definition.
In other words, we have
$$|f(n)| \geq c|g(n)| \text{ if and only if } \quad |g(n)| \leq \frac{1}{c}|f(n)|.$$

Big Theta

- A function f has the **same growth rate as** g, or f has the **same order as** g, if we can find a number m and two positive constants c and d such that

$$c|g(n)| \leq |f(n)| \leq d|g(n)| \text{ for } n \geq m.$$

- In this case we write $f(n) = \Theta(g(n))$ and say that $f(n)$ **is big theta of** $g(n)$.
- If $f(n) = \Theta(g(n))$ and we also know that $g(n) \neq 0$ for all $n \geq m$, then we can divide the inequality in the definition by $g(n)$ to obtain

$$c \leq \left|\frac{f(n)}{g(n)}\right| \leq d \text{ for } n \geq m.$$

- This inequality gives us a better way to think about "having the same growth rate": It says that the ratio of the two functions is always within a fixed bound beyond some point.

Big Theta is an Equivalence Relation

- The relation "has the same growth rate as" between functions is an equivalence relation, i.e., satisfies reflexivity, symmetry and transitivity:
  - $f(n) = \Theta(f(n))$.
  - If $f(n) = \Theta(g(n))$, then $g(n) = \Theta(f(n))$.
  - If $f(n) = \Theta(g(n))$ and $g(n) = \Theta(h(n))$, then $f(n) = \Theta(h(n))$.

Big Theta and Proportionality

- Two functions f and g are **proportional** if there exists a constant a, such that
$$f(n) = ag(n), \text{ for all } n.$$
- If two functions f and g are proportional, then $f(n) = \Theta(g(n))$.
Indeed, if $f(n) = ag(n)$, we have
$$|a||g(n)| \leq |f(n)| \leq |a||g(n)|, \text{ for all } n.$$

Example: Logs Over Different Bases

- Recall the change-of-base formula for logarithms over bases $a > 1$ and $b > 1$:

$$\log_a n = \frac{\log_b n}{\log_b a} \quad \Rightarrow \quad \log_b n = (\log_b a)(\log_a n).$$

- This shows that log functions with different bases are proportional.
- Therefore, we have

$$\log_a n = \Theta(\log_b n).$$

- So we can disregard the base of the log function when considering rates of growth.

Example

- Show that $n^2 + n$ and n^2 have the same growth rate.

  Since $n \leq n^2$, for all n, we have

  $$1 \cdot n^2 \leq n^2 + n \leq n^2 + n^2 = 2 \cdot n^2, \text{ for all } n \geq 1.$$

  We conclude $n^2 + n = \Theta(n^2)$.

Example: Harmonic Numbers

- The nth harmonic number H_n was defined as the sum $H_n = \sum_{k=1}^{n} \frac{1}{k} = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$.
- We have the following bounds for H_n:

$$\ln\left(\frac{n+1}{2}\right) + 1 \leq H_n \leq \ln(n) + 1.$$

Note the following inequalities:

- For $n \geq 3$:

$$\ln(n) + 1 \leq \ln n + \ln n = 2\ln n.$$

- For all n,

$$\ln\left(\frac{n+1}{2}\right) + 1 = \ln\left(\frac{n+1}{2}\right) + \ln e = \ln\left(\frac{e}{2}(n+1)\right) \geq \ln(n+1) > \ln n.$$

So for $n > 3$, $\ln n < H_n < 2\ln n$.

Therefore, $H_n = \Theta(\ln n) = \Theta(\log n)$ since all logarithms have the same rate of growth.

A Limit Test

- To show that two functions have the same growth rate we may use the following test:

  If $\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$, where $c \neq 0$ and $c \neq \infty$, then $f(n) = \Theta(g(n))$.

- Example: Show that if $p(n)$ is a polynomial of degree m, then $p(n) = \Theta(n^m)$.

  Let $p(n) = a_0 + a_1 n + \cdots + a_m n^m$.

  Then we have the following limit:

$$\begin{array}{rcl} \lim_{n\to\infty} \frac{p(n)}{n^m} & = & \lim_{n\to\infty} \frac{a_0 + a_1 n + \cdots + a_m n^m}{n^m} \\ & = & \lim_{n\to\infty} \left(\frac{a_0}{n^m} + \frac{a_1}{n^{m-1}} + \frac{a_2}{n^{m-2}} + \cdots + \frac{a_m}{1}\right) \\ & = & a_m. \end{array}$$

  Since $p(n)$ has degree m, $a_m \neq 0$.

  So by the limit test we have $p(n) = \Theta(n^m)$.

Remark on the Limit Test

- The limit is not a necessary condition for $f(n) = \Theta(g(n))$.
- Example: Consider the functions $f(n)$ and $g(n)$, defined by

$$\begin{array}{rcl} f(n) & = & \text{if } n \text{ is odd then 2 else 4;} \\ g(n) & = & 2. \end{array}$$

Note that

$$\frac{f(n)}{g(n)} = \begin{cases} 1, & \text{if } n \text{ is odd} \\ 2, & \text{otherwise} \end{cases}$$

Therefore, the limit $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ does not exist.

On the other hand, for all n,

$$g(n) \leq f(n) \leq 2g(n).$$

So, we do have $f(n) = \Theta(g(n))$.

Rate of Growth of Finite Sums

- We have the following rates of growth:
  - $\sum_{k=1}^{n} k = \Theta(n^2)$;
  - $\sum_{k=1}^{n} k^2 = \Theta(n^3)$;
  - $\sum_{k=0}^{n} a^k = \Theta(a^{n+1})$, $a \neq 1$.
  - $\sum_{k=1}^{n} ka^k = \Theta(na^{n+1})$, $a \neq 1$.
- More generally we have, for any $r \neq -1$,

$$\sum_{k=1}^{\infty} k^r = \Theta(n^{r+1}).$$

Example

- Show that $\log(n!) = \Theta(n \log n)$.

$$\begin{array}{rcl}
\log(n!) & = & \log n + \log(n-1) + \cdots + \log 1 \\
& \leq & \log n + \log n + \cdots + \log n \\
& = & n \log n; \\
\log(n!) & = & \log n + \log(n-1) + \cdots + \log 1 \\
& \geq & \log n + \log(n-1) + \cdots + \log \lceil \frac{n}{2} \rceil \\
& \geq & \log \lceil \frac{n}{2} \rceil + \log \lceil \frac{n}{2} \rceil + \cdots + \log \lceil \frac{n}{2} \rceil \\
& = & \lceil \frac{n}{2} \rceil \log \lceil \frac{n}{2} \rceil \\
& \geq & \frac{n}{2} \log(\frac{n}{2}).
\end{array}$$

- We conclude $\frac{n}{2} \log(\frac{n}{2}) \leq \log(n!) \leq n \log n$.
- But one can also show that $\frac{1}{2} \log n < \log(\frac{n}{2})$, for all $n > 4$.
- So we get

$$\frac{1}{2} \cdot n \log n \leq \log(n!) \leq 1 \cdot n \log n, \text{ for all } n > 4.$$

Little Oh

- A function f has a **lower growth rate than** g or f has **lower order than** g if
$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=0.$$
- In this case we write $f(n)=o(g(n))$ and say that f **is little oh of** g.
- Example: Show that $n=o(n^2)$.
  We have
$$\lim_{n\to\infty}\frac{n}{n^2}=\lim_{n\to\infty}\frac{1}{n}=0.$$
- Example: Show that if $a,b>0$, such that $a<b$, then $a^n=o(b^n)$.
  We have
$$\lim_{n\to\infty}\frac{a^n}{b^n}=\lim_{n\to\infty}(\frac{a}{b})^n\overset{0<\frac{a}{b}<1}{=}0.$$

Little Oh and L'Hôpital's Rule

- The evaluation of limits can often be accomplished by using L'Hôpital's rule:

  If $\lim_{n\to\infty} f(n) = \lim_{n\to\infty} g(n) = \infty$ or $\lim_{n\to\infty} f(n) = \lim_{n\to\infty} g(n) = 0$ and f and g are differentiate beyond some point, then $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{f'(n)}{g'(n)}$.

- Example: Show that $\log n = o(n)$.

  We have $\lim_{n\to\infty} \log n = \lim_{n\to\infty} n = \infty$.

  Therefore, by L'Hôpital's Rule:

$$\lim_{n\to\infty} \frac{\log n}{n} = \lim_{n\to\infty} \frac{\frac{1}{n \ln 10}}{1} = \lim_{n\to\infty} \frac{1}{n \ln 10} = 0.$$

  So $\log n = o(n)$.

Hierarchy of Growth Rates

- We write $f(n) \prec g(n)$ to mean that $f(n) = o(g(n))$.
- A hierarchy of some familiar functions according to their growth rates:

$$1 \prec \log n \prec n \prec n \log n \prec n^2 \prec n^3 \prec 2^n \prec 3^n \prec n! \prec n^n.$$

- This hierarchy helps in comparing different algorithms.
- Example: We would certainly choose an algorithm with running time $\Theta(\log n)$ over an algorithm with running time $\Theta(n)$.

Using the Symbols in Arithmetic Expressions

- The four symbols O, Ω, Θ and o can also be used to represent terms within an expression.
- Example: The equation

$$h(n) = 4n^3 + O(n^2)$$

means that $h(n)$ equals $4n^3$ plus a term of order at most n^2.

Using the Symbols to Represent Sets of Functions

- The four symbols O, Ω, Θ and o can be formally defined to represent sets of functions:

$O(g)$	The set of functions of order bounded above by that of g
$\Omega(g)$	The set of functions of order bounded below by that of g
$\Theta(g)$	The set of functions with the same order as g
$o(g)$	The set of functions with lower order than g

- Now $f(n) \in O(g(n))$ means that $f(n) = O(g(n))$.
- We may also express set theoretic relations, such as:
  - $O(g(n)) \supseteq \Theta(g(n)) \cup o(g(n))$;
  - $\Theta(g(n)) = O(g(n)) \cap \Omega(g(n))$.