Discrete Mathematics

Topic 05 — Enumeration

Lecture 02 — Binomial Coefficients

Dr Kieran Murphy

Department of Computing and Mathematics,
SETU (Waterford).
(kieran.murphy@setu.ie)

Autumn Semester, 2025/26

Outline

- Counting bit strings
- Counting paths in latices
- Expanding expressions of the form $(a+b)^n$
- Binomial coefficients and Pascal's triangle

Outline

Motivation

Today we will focus on problems that can modelled using repeated/multiple choices where each choice has only two options. Having fewer options should make life easier, but despite this limitation the use of repeated/multiple two-option choices allows us to model a wide variety of problems.

Subsets of a Set

How many subsets of a given set exist that satisfy a particular property?

Bit Strings

How many string containing 0's or 1's of length n satisfy a particular property?

Lattice Paths

Given a grid (think of Manhattan Island, New York) how many shortest paths satisfy a particular property?

Binomial Coefficients

What is the expansion of $(x+y)^n$ for $n \in \mathbb{N}$?

Notation — Factorial Function

Definition 1 (Factorial)

If $n \in \mathbb{N}$ then the factorial function is defined by

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \\ \text{not defned} & \text{otherwise} \end{cases}$$

i.e., the factorial of zero is one, and the factorial of a positive integer, n, is the product of positive integers up to and including n.

The factorial function satisfies

- $n! = n \times (n-1)!$ $\quad 2! = 2 \times 1!, \quad 3! = 3 \times 2!, \quad 4! = 4 \times 3!, \ldots$
- $(n+1)! = (n+1) \times n!$ $\quad (0+1)! = (0+1) \times 0! \implies 1 = 1! = 0!$
- If $r \leq n$ then

$$\frac{n!}{r!} = \frac{n(n-1)\cdots(r+1)\cancel{(r)(r-1)\cdots}(3)(2)(1)}{\cancel{(r)(r-1)\cdots}(3)(2)(1)} = n(n-1)(n-2)\cdots(r+1)$$

Notation — Factorial Function

Definition 1 (Factorial)

If $n \in \mathbb{N}$ then the factorial function is defined by

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \\ \text{not defned} & \text{otherwise} \end{cases}$$

i.e., the factorial of zero is one, and the factorial of a positive integer, n, is the product of positive integers up to and including n.

The factorial function satisfies

- $n! = n \times (n-1)!$ $\quad 2! = 2 \times 1!, \quad 3! = 3 \times 2!, \quad 4! = 4 \times 3!, \ldots$
- $(n+1)! = (n+1) \times n!$ $\quad (0+1)! = (0+1) \times 0! \quad \Longrightarrow \quad 1 = 1! = 0!$
- If $r \leq n$ then

$$\frac{n!}{r!} = \frac{n(n-1)\cdots(r+1)\cancel{(r)(r-1)\cdots(3)(2)(1)}}{\cancel{(r)(r-1)\cdots(3)(2)(1)}} = n(n-1)(n-2)\cdots(r+1)$$

Notation — Factorial Function

Definition 1 (Factorial)

If $n \in \mathbb{N}$ then the factorial function is defined by

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \\ \text{not defned} & \text{otherwise} \end{cases}$$

i.e., the factorial of zero is one, and the factorial of a positive integer, n, is the product of positive integers up to and including n.

The factorial function satisfies

- $n! = n \times (n-1)!$ $\qquad 2! = 2 \times 1!, \quad 3! = 3 \times 2!, \quad 4! = 4 \times 3!, \ldots$
- $(n+1)! = (n+1) \times n!$ $\qquad (0+1)! = (0+1) \times 0! \quad \Longrightarrow \quad 1 = 1! = 0!$
- If $r \leq n$ then

$$\frac{n!}{r!} = \frac{n(n-1)\cdots(r+1)\cancel{(r)(r-1)\cdots(3)(2)(1)}}{\cancel{(r)(r-1)\cdots(3)(2)(1)}} = n(n-1)(n-2)\cdots(r+1)$$

Notation — Factorial Function

Definition 1 (Factorial)

If $n \in \mathbb{N}$ then the factorial function is defined by

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \\ \text{not defned} & \text{otherwise} \end{cases}$$

i.e., the factorial of zero is one, and the factorial of a positive integer, n, is the product of positive integers up to and including n.

The factorial function satisfies

- $n! = n \times (n-1)!$ $\qquad 2! = 2 \times 1!, \quad 3! = 3 \times 2!, \quad 4! = 4 \times 3!, \ldots$
- $(n+1)! = (n+1) \times n!$ $\qquad (0+1)! = (0+1) \times 0! \quad \Longrightarrow \quad 1 = 1! = 0!$
- If $r \leq n$ then

$$\frac{n!}{r!} = \frac{n(n-1)\cdots(r+1)\cancel{(r)(r-1)\cdots(3)(2)(1)}}{\cancel{(r)(r-1)\cdots(3)(2)(1)}} = n(n-1)(n-2)\cdots(r+1)$$

Python Implantation of Factorial — Iterative vs Recursive

We have two different definitions of the factorial function which give us two different implementions

Iterative Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \end{cases}$$

Recursive Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ n \times (n-1)! & n > 0 \end{cases}$$

Python Implantation of Factorial — Iterative vs Recursive

We have two different definitions of the factorial function which give us two different implementions

Iterative Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \end{cases}$$

Recursive Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ n \times (n-1)! & n > 0 \end{cases}$$

factorial_iterative .py

```python
def factorial (n):
    result = 1

    for k in range(1,n+1):
        result *= k
    return result

for k in [0,1,5]:
    print(k,"! =", factorial(k))
```

factorial_recursive .py

```python
def factorial(n):
    # terminal rule
    if n==0: return 1

    # recurrence rule
    return n*factorial(n-1)

for k in [0,1,5]:
    print(k,"! =", factorial(k))
```

Python Implantation of Factorial — Iterative vs Recursive

We have two different definitions of the factorial function which give us two different implementions

Iterative Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ 1 \times 2 \times \cdots \times n & n > 0 \end{cases}$$

Recursive Definition

The factorial of integer n is

$$n! = \begin{cases} 1 & n = 0 \\ n \times (n-1)! & n > 0 \end{cases}$$

```
0 ! = 1
1 ! = 1
5 ! = 120
```

factorial_iterative

```python
def factorial (n):
    result = 1

    for k in range(1,n+1):
        result *= k
    return result

for k in [0,1,5]:
    print(k,"! =", factorial(k))
```

factorial_recursive .py

```python
def factorial(n):
    # terminal rule
    if n==0: return 1

    # recurrence rule
    return n*factorial(n-1)

for k in [0,1,5]:
    print(k,"! =", factorial(k))
```

Counting Subsets I

Problem: Counting Subsets

Given set A determine the number of subsets that satisfy a given criteria.

We want to represent the construction of subsets in terms of multiple, two-option (yes-no) choices:

- If A is a set of $n = |A|$ elements, then any subset of A can be constructed via the algorithm:

$$\left.\begin{array}{ll} \text{For each element } a \text{ in } A: & \\ \quad \text{Decide whether to add } a \text{ to subset or not} & \} \ 2 \text{ options} \end{array}\right\} 2^n \text{options}$$

Example

Given the set of positive integers less than 21, construct the subset containing primes

$$\begin{array}{lcccccccccccccccccccc} A = \{ & 1, & 2, & 3, & 4, & 5, & 6, & 7, & 8, & 9, & 10, & 11, & 12, & 13, & 14, & 15, & 16, & 17, & 18, & 19, & 20 \} \\ & \text{N} & \text{Y} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{N} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{N} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} \\ & & \downarrow & \downarrow & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow & \\ & & \{2, & 3, & & 5, & & 7, & & & & 11, & & 13, & & & & 17, & & 19\} & \end{array}$$

Counting Subsets I

Problem: Counting Subsets

Given set A determine the number of subsets that satisfy a given criteria.

We want to represent the construction of subsets in terms of multiple, two-option (yes-no) choices:

- If A is a set of $n = |A|$ elements, then any subset of A can be constructed via the algorithm:

For each element a in A:
  Decide whether to add a to subset or not $\}$ 2 options $\}$ 2^noptions

Example

Given the set of positive integers less than 21, construct the subset containing primes

$A =$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	N	Y	Y	N	Y	N	Y	N	N	N	Y	N	Y	N	N	N	Y	N	Y	N
		2	3		5		7				11		13				17		19	

$\{2, 3, 5, 7, 11, 13, 17, 19\}$

Counting Subsets I

Problem: Counting Subsets

Given set A determine the number of subsets that satisfy a given criteria.

We want to represent the construction of subsets in terms of multiple, two-option (yes-no) choices:

- If A is a set of $n = |A|$ elements, then any subset of A can be constructed via the algorithm:

$$\left.\begin{array}{l} \text{For each element } a \text{ in } A\text{:} \\ \qquad \text{Decide whether to add } a \text{ to subset or not} \quad \} \text{ 2 options} \end{array}\right\} 2^n \text{options}$$

Example

Given the set of positive integers less than 21, construct the subset containing primes

$$\begin{array}{cccccccccccccccccccccc} A = \{ & 1, & 2, & 3, & 4, & 5, & 6, & 7, & 8, & 9, & 10, & 11, & 12, & 13, & 14, & 15, & 16, & 17, & 18, & 19, & 20 & \} \\ & \text{N} & \text{Y} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{N} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} & \text{N} & \text{N} & \text{Y} & \text{N} & \text{Y} & \text{N} & \\ & & \downarrow & \downarrow & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow & & \\ & & \{2, & 3, & & 5, & & 7, & & & & 11, & & 13, & & & & 17, & & 19\} & & \end{array}$$

Counting Subsets II

Example 2

Consider the set $A = \{1, 2, 3, 4, 5\}$.

(a) How many subsets does A have?

(b) How many of these subsets contain exactly 3 elements?

(a) *How many subsets does A have?*
Number of subsets of A is easy … $|A| = 5$, so number of subsets is $2^5 = 32$.

(b) *How many of these subsets contain exactly 3 elements?*
Using the multiplication principle, to build a subset containing 3 elements …

- We have 5 options for the first element in the subset
- Then, we have 4 options for the second element in the subset.
- Then, we have 3 options for the third element in the subset.

This gives us $5 \times 4 \times 3 = 60$. But this is > 32 (from above) so something is wrong.

Counting Subsets II

Example 2

Consider the set $A = \{1, 2, 3, 4, 5\}$.

(a) How many subsets does A have?

(b) How many of these subsets contain exactly 3 elements?

(a) *How many subsets does A have?*
Number of subsets of A is easy … $|A| = 5$, so number of subsets is $2^5 = 32$.

(b) *How many of these subsets contain exactly 3 elements?*
Using the multiplication principle, to build a subset containing 3 elements …

- We have 5 options for the first element in the subset
- Then, we have 4 options for the second element in the subset.
- Then, we have 3 options for the third element in the subset.

This gives us $5 \times 4 \times 3 = 60$. But this is > 32 (from above) so something is wrong.

Counting Subsets II

Example 2

Consider the set $A = \{1, 2, 3, 4, 5\}$.

(a) How many subsets does A have?

(b) How many of these subsets contain exactly 3 elements?

(a) *How many subsets does A have?*
Number of subsets of A is easy ... $|A| = 5$, so number of subsets is $2^5 = 32$.

(b) *How many of these subsets contain exactly 3 elements?*
Using the multiplication principle, to build a subset containing 3 elements ...

- We have 5 options for the first element in the subset
- Then, we have 4 options for the second element in the subset.
- Then, we have 3 options for the third element in the subset.

This gives us $5 \times 4 \times 3 = 60$. But this is > 32 (from above) so something is wrong.

Counting Subsets II

Example 2

Consider the set $A = \{1, 2, 3, 4, 5\}$.

(a) How many subsets does A have?

(b) How many of these subsets contain exactly 3 elements?

(a) *How many subsets does A have?*
Number of subsets of A is easy ... $|A| = 5$, so number of subsets is $2^5 = 32$.

(b) *How many of these subsets contain exactly 3 elements?*
Using the multiplication principle, to build a subset containing 3 elements ...

- We have 5 options for the first element in the subset
- Then, we have 4 options for the second element in the subset.
- Then, we have 3 options for the third element in the subset.

This gives us $5 \times 4 \times 3 = 60$. But this is > 32 (from above) so something is wrong.

We forgot to take account of the fact that order doesn't matter in a set, so what we have counted here is number of tuples/lists not number of sets.

Counting Subsets III

We need to remove the duplication due to the order not being important — how many duplicates exist?

- In other words, given a set of 3 elements, how many different ordering do we have?
  - We have 3 options for the first element.
  - Then, we have 2 options for the second element.
  - Then, we have 1 option for the third element.

  Giving, us $3 \times 2 \times 1 = 6$ options for the different ordering of the selected 3 elements.
- Hence, the number of subsets of size 3 is $60/6 = 10$

Counting Subsets III

We need to remove the duplication due to the order not being important — how many duplicates exist?

- In other words, given a set of 3 elements, how many different ordering do we have?
  - We have 3 options for the first element.
  - Then, we have 2 options for the second element.
  - Then, we have 1 option for the third element.

  Giving, us $3 \times 2 \times 1 = 6$ options for the different ordering of the selected 3 elements.
- Hence, the number of subsets of size 3 is $60/6 = 10$

Counting Subsets III

We need to remove the duplication due to the order not being important — how many duplicates exist?

- In other words, given a set of 3 elements, how many different ordering do we have?
  - We have 3 options for the first element.
  - Then, we have 2 options for the second element.
  - Then, we have 1 option for the third element.

  Giving, us $3 \times 2 \times 1 = 6$ options for the different ordering of the selected 3 elements.
- Hence, the number of subsets of size 3 is $60/6 = 10$

Counting Subsets III

We need to remove the duplication due to the order not being important — how many duplicates exist?

- In other words, given a set of 3 elements, how many different ordering do we have?
    - We have 3 options for the first element.
    - Then, we have 2 options for the second element.
    - Then, we have 1 option for the third element.

  Giving, us $3 \times 2 \times 1 = 6$ options for the different ordering of the selected 3 elements.
- Hence, the number of subsets of size 3 is $60/6 = 10$

Note that

$$\frac{\text{number of lists}}{\text{number of different orderings}} = \frac{60}{6} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} = \frac{5 \times 4 \times 3 \times (2 \times 1)}{(3 \times 2 \times 1)(2 \times 1)} = \frac{5!}{3!2!}$$

Example 3 I

Example 3

How many subsets of size $0, 1, 2, \ldots, 5$ does $A = \{1, 2, 3, 4, 5\}$ have?

Size 0: The only subset of zero size is the empty set, $\emptyset$, Hence answer is 1. $= \dfrac{5!}{0!5!}$

Size 1: Number of subsets of size one is 5, i.e., $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ and $\{5\}$. Rather than listing and counting them we can use the same argument as in the previous example.

$$\frac{\text{number of lists of length 1}}{\text{number of different orderings}} = \frac{5}{1} = \frac{5!}{1!4!}$$

Size 2:

$$\frac{\text{number of lists of length 2}}{\text{number of different orderings}} = \frac{5 \times 4}{2 \times 1} = \frac{5!}{2!3!}$$

Size 3:

$$\frac{\text{number of lists of length 3}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} = \frac{5!}{3!2!}$$

Example 3 I

Example 3

How many subsets of size $0, 1, 2, \ldots, 5$ does $A = \{1, 2, 3, 4, 5\}$ have?

Size 0: The only subset of zero size is the empty set, $\emptyset$, Hence answer is 1. $= \frac{5!}{0!5!}$

Size 1: Number of subsets of size one is 5, i.e., $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ and $\{5\}$. Rather than listing and counting them we can use the same argument as in the previous example.

$$\frac{\text{number of lists of length 1}}{\text{number of different orderings}} = \frac{5}{1} = \frac{5!}{1!4!}$$

Size 2:

$$\frac{\text{number of lists of length 2}}{\text{number of different orderings}} = \frac{5 \times 4}{2 \times 1} = \frac{5!}{2!3!}$$

Size 3:

$$\frac{\text{number of lists of length 3}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} = \frac{5!}{3!2!}$$

Example 3

Example 3

How many subsets of size $0, 1, 2, \ldots, 5$ does $A = \{1, 2, 3, 4, 5\}$ have?

Size 0: The only subset of zero size is the empty set, $\emptyset$, Hence answer is 1. $= \dfrac{5!}{0!5!}$

Size 1: Number of subsets of size one is 5, i.e., $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ and $\{5\}$. Rather than listing and counting them we can use the same argument as in the previous example.

$$\frac{\text{number of lists of length 1}}{\text{number of different orderings}} = \frac{5}{1} = \frac{5!}{1!4!}$$

Size 2:

$$\frac{\text{number of lists of length 2}}{\text{number of different orderings}} = \frac{5 \times 4}{2 \times 1} = \frac{5!}{2!3!}$$

Size 3:

$$\frac{\text{number of lists of length 3}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} = \frac{5!}{3!2!}$$

Example 3 I

Example 3

How many subsets of size $0, 1, 2, \ldots, 5$ does $A = \{1, 2, 3, 4, 5\}$ have?

Size 0: The only subset of zero size is the empty set, $\emptyset$, Hence answer is 1. $= \frac{5!}{0!5!}$

Size 1: Number of subsets of size one is 5, i.e., $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ and $\{5\}$. Rather than listing and counting them we can use the same argument as in the previous example.

$$\frac{\text{number of lists of length 1}}{\text{number of different orderings}} = \frac{5}{1} \qquad = \frac{5!}{1!4!}$$

Size 2:

$$\frac{\text{number of lists of length 2}}{\text{number of different orderings}} = \frac{5 \times 4}{2 \times 1} \qquad = \frac{5!}{2!3!}$$

Size 3:

$$\frac{\text{number of lists of length 3}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} \qquad = \frac{5!}{3!2!}$$

Example 3 I

Example 3

How many subsets of size $0, 1, 2, \ldots, 5$ does $A = \{1, 2, 3, 4, 5\}$ have?

Size 0: The only subset of zero size is the empty set, $\emptyset$, Hence answer is 1. $= \dfrac{5!}{0!5!}$

Size 1: Number of subsets of size one is 5, i.e., $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$ and $\{5\}$. Rather than listing and counting them we can use the same argument as in the previous example.

$$\frac{\text{number of lists of length 1}}{\text{number of different orderings}} = \frac{5}{1} \qquad = \frac{5!}{1!4!}$$

Size 2:

$$\frac{\text{number of lists of length 2}}{\text{number of different orderings}} = \frac{5 \times 4}{2 \times 1} \qquad = \frac{5!}{2!3!}$$

Size 3:

$$\frac{\text{number of lists of length 3}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3}{3 \times 2 \times 1} \qquad = \frac{5!}{3!2!}$$

Example 3 II

Size 4:

$$\frac{\text{number of lists of length 4}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2}{4 \times 3 \times 2 \times 1} = \frac{5!}{4!1!}$$

Size 5:

$$\frac{\text{number of lists of length 5}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2 \times 1}{5 \times 4 \times 3 \times 2 \times 1} = \frac{5!}{5!0!}$$

Example 3 II

Size 4:

$$\frac{\text{number of lists of length 4}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2}{4 \times 3 \times 2 \times 1} = \frac{5!}{4!1!}$$

Size 5:

$$\frac{\text{number of lists of length 5}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2 \times 1}{5 \times 4 \times 3 \times 2 \times 1} = \frac{5!}{5!0!}$$

Example 3 II

Size 4:

$$\frac{\text{number of lists of length 4}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2}{4 \times 3 \times 2 \times 1} = \frac{5!}{4!1!}$$

Size 5:

$$\frac{\text{number of lists of length 5}}{\text{number of different orderings}} = \frac{5 \times 4 \times 3 \times 2 \times 1}{5 \times 4 \times 3 \times 2 \times 1} = \frac{5!}{5!0!}$$

So we have

Number of elements	0	1	2	3	4	5
Number of subsets	$\frac{5!}{0!5!}$	$\frac{5!}{1!4!}$	$\frac{5!}{2!3!}$	$\frac{5!}{3!2!}$	$\frac{5!}{4!1!}$	$\frac{5!}{5!0!}$
	1	5	10	10	5	1

Subsets

	Subsets
Start	Set with n elements
Construction	For each of the n elements, choice is no-yes to include it in subset
Questions	Number of subsets
	Number of subsets with k elements

Binary String a.k.a. Bit Strings

We will now look at counting bit-strings, where "Bit" is short for "binary digit,"

Definition 4 (Bit String)

A string of of binary digits (0's or 1's) is called a bit string.

- The number of bits in the string is the length of the string. A bit string of length n is often called a n-bit string.
- The number of 1's in a bit string is the weight of the string.

Example

All of the following are bit strings:

$$1001 \quad 0 \quad 1111 \quad 1010101010$$

- The strings above have lengths 4, 1, 4, and 10 respectively.
- The weights of the above strings are 2, 0, 4, and 5 respectively.
- The construction of a n-bit can be modelled by n yes-no choices.

Construction of bit-strings is similar to subsets, but here order matters

Binary String a.k.a. Bit Strings

We will now look at counting bit-strings, where "Bit" is short for "binary digit,"

Definition 4 (Bit String)

A string of of binary digits (0's or 1's) is called a bit string.

- The number of bits in the string is the length of the string. A bit string of length n is often called a n-bit string.
- The number of 1's in a bit string is the weight of the string.

Example

All of the following are bit strings:

$$1001 \quad 0 \quad 1111 \quad 1010101010$$

- The strings above have lengths 4, 1, 4, and 10 respectively.
- The weights of the above strings are 2, 0, 4, and 5 respectively.
- The construction of a n-bit can be modelled by n yes-no choices.

Construction of bit-strings is similar to subsets, but here order matters

Counting Bit Strings

Since the construction of bit strings is similar to the construction of subsets we can ask similar questions such as

- How many bit strings of length n exist?
- How many bit strings of length n have weight k?

However, since order is important for bit strings we can also ask questions like

- How many bit strings start/end/contain a specified sequence of bits.
- How many n-bit strings satisfy a specified numerical property?

Notation

- $\mathbf{B}^n$ is the set of all bit strings of length n.
- $\mathbf{B}^n_k$ is the set of all bit strings of length n and weight k.

Example

The set $\mathbf{B}^3_2$ represents strings containing three bits exactly two of which are 1'a:

$$\mathbf{B}^3_2 = \{011, 101, 110\}.$$

and the number of 3-bit strings with weight 2 is $|\mathbf{B}^3_2|$

Counting Bit Strings

Since the construction of bit strings is similar to the construction of subsets we can ask similar questions such as

- How many bit strings of length n exist?
- How many bit strings of length n have weight k?

However, since order is important for bit strings we can also ask questions like

- How many bit strings start/end/contain a specified sequence of bits.
- How many n-bit strings satisfy a specified numerical property?

Notation

- $\mathbf{B}^n$ is the set of all bit strings of length n.
- $\mathbf{B}^n_k$ is the set of all bit strings of length n and weight k.

Example

The set $\mathbf{B}^3_2$ represents strings containing three bits exactly two of which are 1'a:

$$\mathbf{B}^3_2 = \{011, 101, 110\}.$$

and the number of 3-bit strings with weight 2 is $|\mathbf{B}^3_2|$

Example 5

Example 5

What is the cardinality of $\mathbf{B}^5$, i.e., how many 5-bit string are there?

Solution: Each of the 5 bits can either be a 0 or a 1, i.e., 5 yes-no choices. There are 2 choices for the first bit, 2 choices for the second, and so on. By the multiplication principle, there are $2 \times 2 \times 2 \times 2 \times 2 = 2^5 = 32$ such strings, so $|\mathbf{B}^5| = 32$.

Note: We can also answer this question using recursion — expressing aproblem in terms of a simpler/smaller version of itself.

- All 5-bit strings can be constructed from prepending either a 0 or a 1 to all of the 4-bit strings.

$$\mathbf{B}^5 = \underbrace{\{0b | b \in \mathbf{B}^4\}}_{00000,00001,\ldots,01111} \cup \underbrace{\{1b | b \in \mathbf{B}^4\}}_{10000,10001,\ldots,11111} \implies |\mathbf{B}^5| = 2|\mathbf{B}^4|$$

- Similarly all 4-bit strings are constructed from 3-bit strings, and so on with $|\mathbf{B}^1| = 2$.

Example 5

Example 5

What is the cardinality of $\mathbf{B}^5$, i.e., how many 5-bit string are there?

Solution: Each of the 5 bits can either be a 0 or a 1, i.e., 5 yes-no choices. There are 2 choices for the first bit, 2 choices for the second, and so on. By the multiplication principle, there are $2 \times 2 \times 2 \times 2 \times 2 = 2^5 = 32$ such strings, so $|\mathbf{B}^5| = 32$.

Note: We can also answer this question using recursion — expressing aproblem in terms of a simpler/smaller version of itself.

- All 5-bit strings can be constructed from prepending either a 0 or a 1 to all of the 4-bit strings.

$$\mathbf{B}^5 = \underbrace{\{0b \mid b \in \mathbf{B}^4\}}_{00000,00001,\ldots,01111} \cup \underbrace{\{1b \mid b \in \mathbf{B}^4\}}_{10000,10001,\ldots,11111} \implies |\mathbf{B}^5| = 2|\mathbf{B}^4|$$

- Similarly all 4-bit strings are constructed from 3-bit strings, and so on with $|\mathbf{B}^1| = 2$.

Example 5

Example 5

What is the cardinality of $\mathbf{B}^5$, i.e., how many 5-bit string are there?

Solution: Each of the 5 bits can either be a 0 or a 1, i.e., 5 yes-no choices. There are 2 choices for the first bit, 2 choices for the second, and so on. By the multiplication principle, there are $2 \times 2 \times 2 \times 2 \times 2 = 2^5 = 32$ such strings, so $|\mathbf{B}^5| = 32$.

Note: We can also answer this question using recursion — expressing aproblem in terms of a simpler/smaller version of itself.

- All 5-bit strings can be constructed from prepending either a 0 or a 1 to all of the 4-bit strings.

$$\mathbf{B}^5 = \underbrace{\{0b | b \in \mathbf{B}^4\}}_{00000,00001,\ldots,01111} \cup \underbrace{\{1b | b \in \mathbf{B}^4\}}_{10000,10001,\ldots,11111} \implies |\mathbf{B}^5| = 2|\mathbf{B}^4|$$

- Similarly all 4-bit strings are constructed from 3-bit strings, and so on with $|\mathbf{B}^1| = 2$.

Example 5

Example 5

What is the cardinality of $\mathbf{B}^5$, i.e., how many 5-bit string are there?

Solution: Each of the 5 bits can either be a 0 or a 1, i.e., 5 yes-no choices. There are 2 choices for the first bit, 2 choices for the second, and so on. By the multiplication principle, there are $2 \times 2 \times 2 \times 2 \times 2 = 2^5 = 32$ such strings, so $|\mathbf{B}^5| = 32$.

Note: We can also answer this question using recursion — expressing aproblem in terms of a simpler/smaller version of itself.

- All 5-bit strings can be constructed from prepending either a 0 or a 1 to all of the 4-bit strings.

$$\mathbf{B}^5 = \underbrace{\{0b|b \in \mathbf{B}^4\}}_{00000,00001,\ldots,01111} \cup \underbrace{\{1b|b \in \mathbf{B}^4\}}_{10000,10001,\ldots,11111} \implies |\mathbf{B}^5| = 2|\mathbf{B}^4|$$

- Similarly all 4-bit strings are constructed from 3-bit strings, and so on with $|\mathbf{B}^1| = 2$.

$$|\mathbf{B}^5| = 2\left(|\mathbf{B}^4|\right) = 2\left(2|\mathbf{B}^3|\right) = 2\left(2\left(2|\mathbf{B}^2|\right)\right) = 2\left(2\left(2\left(2|\mathbf{B}^1|\right)\right)\right) = 2^5$$

Example 6

I

Example 6

What is the cardinality of $\mathbf{B}_3^5$, i.e., how many 5-bit strings have weight 3?

To construct a 5-bit string with exactly 3 ones, we can work as in the last example — we prepend a zero or a one to a 4-bit string. However, we need to take care to ensure we end up with exactly 3 ones.

$$\underbrace{\mathbf{B}_3^5}_{\substack{\text{5-bit strings}\\\text{with exactly 3 ones}}} = \{0b \mid b \in \underbrace{\mathbf{B}_3^4}_{\substack{\text{4-bit string}\\\text{with exactly 3 ones}}}\} \cup \{1b \mid b \in \underbrace{\mathbf{B}_2^4}_{\substack{\text{4-bit string}\\\text{with exactly 2 ones}}}\} \implies |B_3^5| = |B_3^4| + |B_2^4|$$

This is a recurrence relation. We can repeat the above argument on $|B_3^4|$ and on $|B_2^4|$ to get

$$|B_3^4| = \underbrace{|B_3^3|}_{B_3^3=\{111\}} + |B_2^3| = 1 + \left(\underbrace{|B_2^2|}_{B_2^2=\{11\}} + |B_1^2|\right) = 1 + 1 + \left(\underbrace{|B_1^1|}_{B_1^1=\{1\}} + \underbrace{|B_0^1|}_{B_0^1=\{0\}}\right) = 4$$

and

$$|B_2^4| = |B_2^3| + |B_1^3| = \left(|B_2^2| + |B_1^2|\right) + \left(|B_1^2| + |B_0^2|\right) = 1 + 2 + 2 + 1 = 6$$

Hence $|B_3^5| = |B_3^4| + |B_2^4| = 4 + 6 = 10.$

Example 6 I

Example 6

What is the cardinality of $\mathbf{B}_3^5$, i.e., how many 5-bit strings have weight 3?

To construct a 5-bit string with exactly 3 ones, we can work as in the last example — we prepend a zero or a one to a 4-bit string. However, we need to take care to ensure we end up with exactly 3 ones.

$$\underbrace{\mathbf{B}_3^5}_{\substack{\text{5-bit strings}\\ \text{with exactly 3 ones}}} = \{0b \mid b \in \underbrace{\mathbf{B}_3^4}_{\substack{\text{4-bit string}\\ \text{with exactly 3 ones}}}\} \cup \{1b \mid b \in \underbrace{\mathbf{B}_2^4}_{\substack{\text{4-bit string}\\ \text{with exactly 2 ones}}}\} \implies |B_3^5| = |B_3^4| + |B_2^4|$$

This is a recurrence relation. We can repeat the above argument on $|B_3^4|$ and on $|B_2^4|$ to get

$$|B_3^4| = \underbrace{|B_3^3|}_{B_3^3 = \{111\}} + |B_2^3| = 1 + \Big(\underbrace{|B_2^2|}_{B_2^2 = \{11\}} + |B_1^2|\Big) = 1 + 1 + \Big(\underbrace{|B_1^1|}_{B_1^1 = \{1\}} + \underbrace{|B_0^1|}_{B_0^1 = \{0\}}\Big) = 4$$

and

$$|B_2^4| = |B_2^3| + |B_1^3| = \big(|B_2^2| + |B_1^2|\big) + \big(|B_1^2| + |B_0^2|\big) = 1 + 2 + 2 + 1 = 6$$

Hence $|B_3^5| = |B_3^4| + |B_2^4| = 4 + 6 = 10$.

Example 6

I

Example 6

What is the cardinality of $\mathbf{B}_3^5$, i.e., how many 5-bit strings have weight 3?

To construct a 5-bit string with exactly 3 ones, we can work as in the last example — we prepend a zero or a one to a 4-bit string. However, we need to take care to ensure we end up with exactly 3 ones.

$$\underbrace{\mathbf{B}_3^5}_{\substack{\text{5-bit strings}\\\text{with exactly 3 ones}}} = \{0b \mid b \in \underbrace{\mathbf{B}_3^4}_{\substack{\text{4-bit string}\\\text{with exactly 3 ones}}}\} \cup \{1b \mid b \in \underbrace{\mathbf{B}_2^4}_{\substack{\text{4-bit string}\\\text{with exactly 2 ones}}}\} \implies |B_3^5| = |B_3^4| + |B_2^4|$$

This is a recurrence relation. We can repeat the above argument on $|B_3^4|$ and on $|B_2^4|$ to get

$$|B_3^4| = \underbrace{|B_3^3|}_{B_3^3=\{111\}} + |B_2^3| = 1 + \Big(\underbrace{|B_2^2|}_{B_2^2=\{11\}} + |B_1^2|\Big) = 1 + 1 + \Big(\underbrace{|B_1^1|}_{B_1^1=\{1\}} + \underbrace{|B_0^1|}_{B_0^1=\{0\}}\Big) = 4$$

and

$$|B_2^4| = |B_2^3| + |B_1^3| = \big(|B_2^2| + |B_1^2|\big) + \big(|B_1^2| + |B_0^2|\big) = 1 + 2 + 2 + 1 = 6$$

Hence $|B_3^5| = |B_3^4| + |B_2^4| = 4 + 6 = 10.$

Example 6

I

Example 6

What is the cardinality of $\mathbf{B}^5_3$, i.e., how many 5-bit strings have weight 3?

To construct a 5-bit string with exactly 3 ones, we can work as in the last example — we prepend a zero or a one to a 4-bit string. However, we need to take care to ensure we end up with exactly 3 ones.

$$\underbrace{\mathbf{B}^5_3}_{\substack{\text{5-bit strings}\\\text{with exactly 3 ones}}} = \{0b | b \in \underbrace{\mathbf{B}^4_3}_{\substack{\text{4-bit string}\\\text{with exactly 3 ones}}}\} \quad \cup \quad \{1b | b \in \underbrace{\mathbf{B}^4_2}_{\substack{\text{4-bit string}\\\text{with exactly 2 ones}}}\} \qquad \Longrightarrow \qquad |B^5_3| = |B^4_3| + |B^4_2|$$

This is a recurrence relation. We can repeat the above argument on $|B^4_3|$ and on $|B^4_2|$ to get

$$|B^4_3| = \underbrace{|B^3_3|}_{B^3_3 = \{111\}} + |B^3_2| = 1 + \left(\underbrace{|B^2_2|}_{B^2_2 = \{11\}} + |B^2_1|\right) = 1 + 1 + \left(\underbrace{|B^1_1|}_{B^1_1 = \{1\}} + \underbrace{|B^1_0|}_{B^1_0 = \{0\}}\right) = 4$$

and

$$|B^4_2| = |B^3_2| + |B^3_1| = \left(|B^2_2| + |B^2_1|\right) + \left(|B^2_1| + |B^2_0|\right) = 1 + 2 + 2 + 1 = 6$$

Hence $|B^5_3| = |B^4_3| + |B^4_2| = 4 + 6 = 10$.

Recurrence Relationship for $|B_k^n|$

The recursive relation we developed in the previous example for $|B_3^5|$ can be generalised to

Recurrence Relation for $|B_k^n|$

$$|B_k^n| = |B_{k-1}^{n-1}| + |B_k^{n-1}|$$

with terminal conditions $|B_n^n| = 1$ and $|B_0^n| = 1$ for all n.

This results come from the facts:

> *All n-bit string with weight k can be constructed by prepending a one to $(n-1)$-bit strings with weight $(k-1)$ or by prepending a zero to $(n-1)$-bit strings with weight k.*

and

> *There is only one n-bit string with weight n, i.e., all 1's. And there is only one n-bit string with weight 0, i.e., all 0's.*

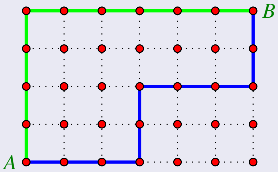
Subsets vs Bit Strings

	Subsets	Bit Strings
Start	Set with n elements	Space for an n-bit string
Construction	For each of the n elements, choice is no-yes to include it in subset	For each of the n bits spaces, choice is 0-1 (or off-on, no-yes).
Questions	Number of subsets	Number of n-bit strings
	Number of subsets with k elements	Number of n-bit strings with weight k

Lattices

Definition 7 (Lattice and Lattice Paths)

A lattice is a 2D grid of points (x, y) where x and y coordinates are integers.
A lattice path is one of the shortest possible paths connecting two points on the lattice, moving only horizontally and vertically.



Problem: Counting Lattice Paths

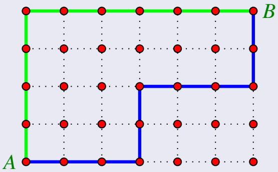
Given start point, A, and end point, B, determine the number of lattice paths that satisfy a given criteria.

- To ensure that a path is one of the shortest possible, each move must be either to the right or up.
- In the above diagram, note that no matter what path we take, we must make four steps up and six steps to the right. No matter what order we make these steps, there will always be 10 steps. Thus each path has *length* 10.
- The construction of a lattice path can be represented as a repeated two option (up-right) choice.

Lattices

Definition 7 (Lattice and Lattice Paths)

A lattice is a 2D grid of points (x, y) where x and y coordinates are integers.
A lattice path is one of the shortest possible paths connecting two points on the lattice, moving only horizontally and vertically.



Problem: Counting Lattice Paths

Given start point, A, and end point, B, determine the number of lattice paths that satisfy a given criteria.

- To ensure that a path is one of the shortest possible, each move must be either to the right or up.
- In the above diagram, note that no matter what path we take, we must make four steps up and six steps to the right. No matter what order we make these steps, there will always be 10 steps. Thus each path has *length* 10.
- The construction of a lattice path can be represented as a repeated two option (up-right) choice.

Example 8

Example 8

How many lattice paths between $A = (0, 0)$ and $B = (3, 2)$?

Three (out of ?) possible lattices parts are

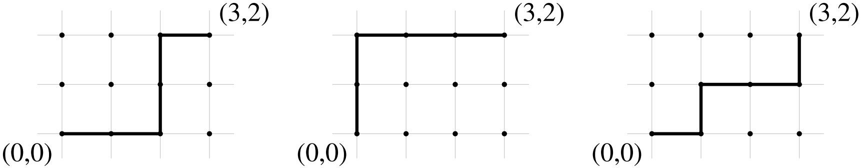
- To ensure the path is the *shortest* possible, each move must be either to the right or up.
- No matter what path we take, we must make three steps right and two steps up.
- No matter what order we make these steps, there will always be 5 steps. Thus each path has *length* 5.
- Rather than drawing a lattice path we could just list which direction (U=up,R=Right) we travel on each of the 5 steps. The lattice paths shown above are RRUUR, UURRR, and RURRU respectively.

Example 8 I

Example 8

How many lattice paths between $A = (0,0)$ and $B = (3,2)$?

Three (out of ?) possible lattices parts are



- To ensure the path is the *shortest* possible, each move must be either to the right or up.
- No matter what path we take, we must make three steps right and two steps up.
- No matter what order we make these steps, there will always be 5 steps. Thus each path has *length* 5.
- Rather than drawing a lattice path we could just list which direction (U=up,R=Right) we travel on each of the 5 steps. The lattice paths shown above are RRUUR, UURRR, and RURRU respectively.

Example 8 II

$$\begin{array}{ccc} \text{RRUUR} & \text{UURRR} & \text{RURRU} \\ \updownarrow\updownarrow\updownarrow\updownarrow\updownarrow & \updownarrow\updownarrow\updownarrow\updownarrow\updownarrow & \updownarrow\updownarrow\updownarrow\updownarrow\updownarrow \\ 1\,1\,0\,0\,1 & 0\,0\,1\,1\,1 & 1\,0\,1\,1\,0 \end{array}$$

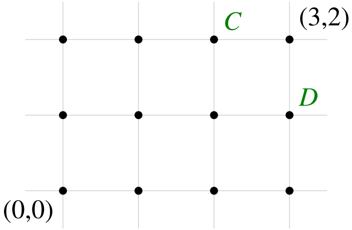
- Notice that each of these strings must contain 5 symbols. Exactly 3 of them must be R's (since our destination is 3 units to the right).
  In fact, say, we used 1's instead of R's and 0's instead of U's? Then we would just have 5-bit strings of weight 3. From previous example, we know that there are 10 of those, so there are 10 lattice paths from (0,0) to (3,2).

Number of lattice paths from $A = (a_x, a_y)$ to $B = (b_x, b_y)$ is equal to the number of n-bit strings with weight k where

$$n = \underbrace{|b_x - a_x|}_{\substack{\text{number of}\\ \text{right moves}}} + \underbrace{|b_y - a_y|}_{\substack{\text{number of}\\ \text{up moves}}} \qquad k = \underbrace{|b_x - a_x|}_{\substack{\text{number of}\\ \text{right moves}}}$$

Example 8 III

There is an another way to count lattice paths. Consider the lattice shown below:



- Any lattice path from (0,0) to (3,2) must pass through exactly one of C and D.
- The point C is 4 steps away from (0,0) and two of them are towards the right. The number of lattice paths to C is the same as the number of 4-bit strings of weight 2, namely 6.
- The point D is 4 steps away from (0,0), but now 3 of them are towards the right. So the number of paths to point B is the same as the number of 4-bit strings of weight 3, namely 4.
- So the total number of paths to (3,2) is just $6 + 4$.
  This is the same way we calculated the number of 5-bit strings of weight 3.

Subsets vs Bit Strings vs Lattice Paths

	Subsets	Bit Strings	Lattice Paths
Start	Set with n elements	Space for an n-bit string	Want a lattice path of length n steps
Construction	For each of the n elements, choice is no-yes to include it in subset	For each of the n bits spaces, choice is 0-1 (or off-on, no-yes).	For each of the n steps, choice is Up-Right (or 0-1).
Questions	Number of subsets	Number of n-bit strings	Number of lattice paths of length n
	Number of subsets with k elements	Number of n-bit strings with weight k	Number of lattice paths of length n with k steps to the right

Binomial Coefficients I

Definition 9 (Binomial Coefficients)

Binomial coefficients are the coefficients in the expanded version of a binomial, such as $(x+y)^n$.

What happens when we multiply such a binomial out? We will expand $(x+y)^n$ for various values of n. Each of these are done by multiplying everything out* and then collecting like terms.

$(x+y)^0 = 1$	1						
$(x+y)^1 = 1x + 1y$	1	1					
$(x+y)^2 = 1x^2 + 2xy + 1y^2$	1	2	1				
$(x+y)^3 = 1x^3 + 3x^2y + 3xy^2 + 1y^3$	1	3	3	1			
$(x+y)^4 = 1x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + 1y^4$	1	4	6	4	1		
$(x+y)^5 = 1x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + 1y^5$	1	5	10	10	5	1	

*In programming terms, this is called FOIL-ing.

Binomial Coefficients II

If we define $\binom{n}{k} = \frac{n!}{k!(n-k)!}$ then the expansion of

$$(x+y)^5 = 1x^5 + 5x^4y + 10x^3y^2 + 10x^2y^3 + 5xy^4 + 1y^5$$

can be written as

$$(x+y)^5 = \binom{5}{0}x^5y^0 + \binom{5}{1}x^4y^1 + \binom{5}{2}x^3y^2 + \binom{5}{3}x^2y^3 + \binom{5}{4}x^1y^4 + \binom{5}{5}x^0y^5$$

In for general power, n, we have $(x+y)^n$ can be written as

Binomial Expansion Theorem

$$(x+y)^n = \binom{n}{0}x^ny^0 + \binom{n}{1}x^{n-1}y^1 + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{n-1}x^1y^{n-1} + \binom{n}{n}x^0y^n$$

Binomial Coefficients III

To summarise what we know about binomial coefficients:

Definition 10 (Binomial Coefficients)

For each integer $n \geq 0$ and integer k with $0 \leq k \leq n$ the number

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{(n)(n-1)(n-2)\cdots(n-k+2)(n-k+1)}{(k)(k-1)(k-2)\cdots(3)(2)(1)}$$

read "n choose k" is a binomial coefficient. We have

- $\binom{n}{k} = |\mathbf{B}_k^n|$, the number of n-bit strings of weight k.
- $\binom{n}{k}$ is the number of subsets of a set of size n each with cardinality k.
- $\binom{n}{k}$ is the number of lattice paths of length n containing k steps to the right.
- $\binom{n}{k}$ is the coefficient of $x^{n-k}y^k$ in the expansion of $(x+y)^n$.

and in general

- $\binom{n}{k}$ is the number of ways to select k objects from a total of n objects.

Binomial Coefficients IV

We can represent the solution to our four applications in terms of $\binom{n}{k}$:

Selecting Subsets

How many subsets of $\{1, 2, 3, 4, 5\}$ contain exactly 3 elements? We must choose 3 of the 5 elements to be in our subset.
There are $\binom{5}{3}$ ways to do this, so there are $\binom{5}{3}$ such subsets.

Bit String

How many bit strings have length 5 and weight 3? We must choose 3 of the 5 bits to be 1's.
There are $\binom{5}{3}$ ways to do this, so there are $\binom{5}{3}$ such bit strings.

Lattice Paths

How many lattice paths are there from (0,0) to (3,2)? We must choose 3 of the 5 steps to be towards the right.
There are $\binom{5}{3}$ ways to do this, so there are $\binom{5}{3}$ such lattice paths.

Binomial Coefficients

What is the coefficient of x^3y^2 in the expansion of $(x+y)^5$?
The coefficient is $\binom{5}{3}$.

Recurrence Relation for Binomial Coefficients

- The number of n-bit strings with weight k, denoted by $|B_k^n|$, is equal to the binomial coefficient $\binom{n}{k}$, and we have a recurrence relation for $|B_k^n|$, i.e.,

$$|B_k^n| = |B_{k-1}^{n-1}| + |B_k^{n-1}|$$

with terminal conditions $|B_n^n| = 1$ and $|B_0^n| = 1$.

- Hence we get the following recurrence relation

<u>Recurrence Relation for Binomial Coefficients)</u>

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

with terminal conditions $\binom{n}{n} = 1$ and $\binom{n}{0} = 1$.

This recurrence relation has a nice geometric interpretation when the binomial coefficients are arranged in a triangle, called Pascal's triangle.

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$\begin{array}{ccc} & 1 & \\ \mathbf{1} & & \mathbf{1} \end{array}$$

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$
\begin{array}{ccccc}
 & & 1 & & \\
 & 1 & & 1 & \\
\mathbf{1} & & & & \mathbf{1}
\end{array}
$$

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$
\begin{array}{ccccc}
 & & 1 & & \\
 & 1 & & 1 & \\
1 & & \mathbf{2} & & 1
\end{array}
$$

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

1 1

1 2 1

1 **1**

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$
\begin{array}{ccccccc}
 & & & 1 & & & \\
 & & 1 & & 1 & & \\
 & 1 & & 2 & & 1 & \\
1 & & \mathbf{3} & & \mathbf{3} & & 1
\end{array}
$$

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$
\begin{array}{ccccccccc}
 & & & & 1 & & & & \\
 & & & 1 & & 1 & & & \\
 & & 1 & & 2 & & 1 & & \\
 & 1 & & 3 & & 3 & & 1 & \\
\mathbf{1} & & & & & & & & \mathbf{1}
\end{array}
$$

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

1 1

1 2 1

1 3 3 1

1 **4** **6** **4** 1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 **1**

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 **5** **10** **10** **5** 1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$n = 0$ 1

$n = 1$ 1 1

$n = 2$ 1 2 1

$n = 3$ 1 3 3 1

$n = 4$ 1 4 6 4 1

$n = 5$ 1 5 10 10 5 1

$n = 6$ 1 6 15 20 15 6 1

$n = 7$ 1 7 21 35 35 21 7 1

$n = 8$ 1 8 28 56 70 56 28 8 1

$n = 9$ 1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$k = 0$, $k = 1$, $k = 2$, $k = 3$, $k = 4$, $k = 5$, $k = 6$, $k = 7$, $k = 8$, $k = 9$

$n = 0$ 1
$n = 1$ 1 1
$n = 2$ 1 2 1
$n = 3$ 1 3 3 1
$n = 4$ 1 4 6 4 1
$n = 5$ 1 5 10 10 5 1
$n = 6$ 1 6 15 20 15 6 1
$n = 7$ 1 7 21 35 35 21 7 1
$n = 8$ 1 8 28 56 70 56 28 8 1
$n = 9$ 1 9 36 84 126 126 84 36 9 1
1 10 45 120 210 252 210 120 45 10 1

Pascal's Triangle

- Start with a 1 — to represent $\binom{0}{0}$
- Place 1 at either end — to represent $\binom{n}{0}$ and $\binom{n}{n}$
- Other entries is sum of upper left and upper right
  $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$

$$\binom{5}{3}$$

	$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$	$k=8$	$k=9$
$n=0$	1									
$n=1$	1	1								
$n=2$	1	2	1							
$n=3$	1	3	3	1						
$n=4$	1	4	6	4	1					
$n=5$	1	5	10	10	5	1				
$n=6$	1	6	15	20	15	6	1			
$n=7$	1	7	21	35	35	21	7	1		
$n=8$	1	8	28	56	70	56	28	8	1	
$n=9$	1	9	36	84	126	126	84	36	9	1

Review Exercises 1 (Binomial Coefficients)

Question 1:
Let $A = \{1, 2, 3, \ldots, 9\}$.

1. How many subsets of A are there? That is, find $|\mathcal{P}(A)|$. Explain.
2. How many subsets of A contain exactly 5 elements? Explain.
3. How many subsets of A contain only even numbers? Explain.
4. How many subsets of A contain an even number of elements? Explain.

Question 2:
How many 9-bit strings (that is, bit strings of length 9) are there which satisfy each of the following criteria? Explain your answers.

1. Start with the sub-string 101.
2. Have weight 5 (i.e., contain exactly five 1's) and start with the sub-string 101.
3. Either start with 101 or end with 11 (or both).
4. Have weight 5, and starts with 101 and ends with 11.

Question 3:
How many shortest lattice paths start at (3,3) and

1. end at (10,10)?
2. end at (10,10) and pass through (5,7)?
3. end at (10,10) and avoid (5,7)?

Question 4:
What is the coefficient of x^{12} in $(x+2)^{15}$?