CHAPTER 5
COUNTING

Outline

- **Content**
  - The basics of counting
  - The pigeonhole principle
- **Reading**
  - Chapter 5

Most of the following slides are by courtesy of Prof. J.-D. Huang and Prof. M.P. Frank

Combinatorics

- **Combinatorics**
  - The study of arrangements of objects
  - Studied since 17th century for gambling
  - Counting / enumeration
- **Objective of counting**
  - The complexity of algorithms
  - There are enough phone numbers / IP addresses
  - 賭場賠率

Basic Counting Principles

- **In this section, we study 2 basic counting principles**
  - product rule
  - sum rule

Product rule

- **Product rule**
  - Suppose that a procedure can be broken down into a sequence of 2 tasks
  - If there are n_1 ways to do the first task and n_2 ways to do the second task after the first task has been done
  - Then there are $n_1 * n_2$ ways to do the procedure

- **Extended product rule**
  - suppose that a procedure can be broken down into a sequence of m tasks, T_1, T_2, …, T_m
  - if there are n_i ways to do the task T_i after Task T_1~$T_{i\text{-}1}$ has been done
  - then there are $n_1 * n_2 * \ldots * n_m$ ways to do the procedure

Simple Examples

- **Labeling chairs with a letter and a positive integer ≤ 100**
- **How many different bit strings are there of length 7?**
- **How many different license plates are available if each plate contains a sequence of 3 letters followed by 3 digits?**
- **How many functions are there from a set with *m* elements to one with *n* elements?**
- **How many one-to-one functions are there from a set with *m* elements to one with *n* elements? ($m \leq n$)**

Telephone Numbering Plan

- **Phone numbering**
  - 10 digits; 3-digit area code, 3-digit office code, 4-digit station code
  - X: 0~9, N: 2~9, Y: 0~1
  - How many available numbers in *NYX-NNX-XXXX* format?
  - How many available numbers in *NXX-NXX-XXXX* format?

Step Counts in Programs

$k := 0$

for $i_1 := 1$ **to** n_1

  for $i_2 := 1$ **to** n_2

    …

      for $i_m := 1$ **to** n_m

        $k := k + 1$

- **Q: What is the value of *k* at last?**

Counting Subsets of a Finite Set

- **Use the product rule to show that the number of different subsets of a finite set *S* is $2^{|S|}$**
  - We have proved this before by mathematical induction
- **Now in different way**

Cartesian Product

- **If $A_1, A_2, \ldots, A_m$ are finite sets, then the number of elements in the Cartesian product of these sets is the product of the number of elements in each set**
- **That is,**
  $|A_1 \times A_2 \times \ldots \times A_m| = |A_1| * |A_2| * \ldots * |A_m|$

Sum Rule

- **The sum rule**
  - If a task can be done either in n_1 ways or in n_2 ways
  - None of the set of n_1 ways is the same as any of the set of n_2 ways
  - Then there are $n_1 + n_2$ ways to do the task

- **The extended sum rule**
  - If the task can be done in $n_1, n_2, \ldots,$ or n_m ways
  - No 2 of these ways are the same
  - Then there are $n_1 + n_2 + \ldots + n_m$ ways to do the task

Simple Examples

- **Either a professor or a student can be selected as a committee member. How many different choices if there are 37 professors and 83 students?**

- **A student can choose a project from 1 of 3 lists. The 3 lists contain 23, 15, and 19 possible projects, respectively. No project is on more than one list. How many possible projects are there to choose from?**

Step Counts in Programs

$k := 0$

for $i_1 := 1$ **to** n_1

    $k := k + 1$

for $i_2 := 1$ **to** n_2

    $k := k + 1$

…

for $i_m := 1$ **to** n_m

    $k := k + 1$

- **Q: What is the value of *k* at last?**

Sum Rule in Disjoint Sets

- **If A_1, A_2, …, A_m are disjoint finite sets, then the number of elements in the union of these sets is the sum of the number of elements in each set**
- **That is,**
  $| A_1 \cup A_2 \cup \ldots \cup A_m | = |A_1| + |A_2| + \ldots + |A_m|$

Number of Available Variable Names

- **For some computer language**
  - The name of a variable is a string of 1 or 2 alphanumeric characters and case-insensitive
  - A variable must begin with a letter
  - 5 2-character strings are reserved for system use
- **How many different variable names are available?**

Passwords

- **Password rule**
  - 6~8 characters long
  - each character is a lowercase letter or a digit
  - must contain at least 1 digit

- **The answer is 2,648,483,063,360**

IP Addresses

- **How many available addresses for Class A, B, and C?**
- **Restrictions**
  - 1111111 cannot be a netid for class A
  - Hostids consisting of all 0s or 1s are not allowed
- **The answer is 3,737,091,842**

IPv4 (Version 4 of the Internet Protocol)

Bit Number	0	1	2	3	4	8	16	24	31
Class A	0	netid				hostid			
Class B	1	0	netid				hostid		
Class C	1	1	0	netid				hostid	
Class D	1	1	1	0	Multicast Address				
Class E	1	1	1	1	0	Address			

Inclusion-Exclusion Principle

- **The sum rule does not work if some ways of 2 tasks are the same**
- **Principle of inclusion-exclusion**
  - Correctly count the number of ways to do 1 of the 2 tasks
  - We add the number of ways to do each of the 2 tasks and then subtract the number of ways to do both tasks

Example

- **How many bit strings of length 8 either start with a 1 bit or end with 2 bits 00?**
  - Starting with 1 : $2^7 = 128$
  - Ending with 00: $2^6 = 64$
  - Starting with 1 & ending with 00: $2^5 = 32$
  - By inclusion-exclusion rule: $128 + 64 – 32 = 160$

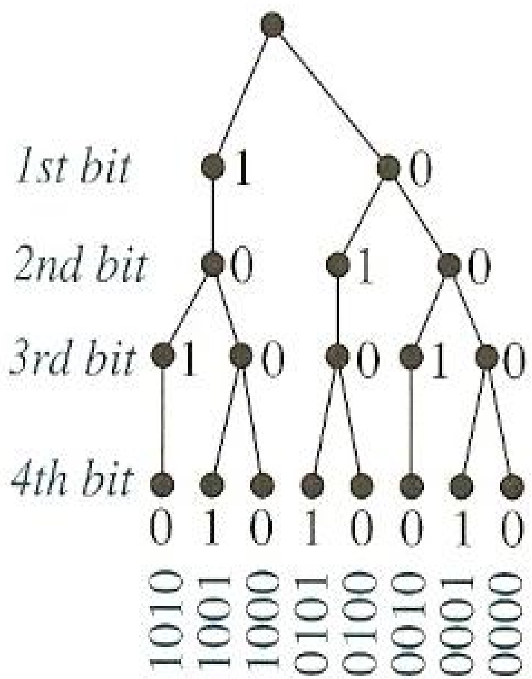
Inclusion-Exclusion in Sets

- **Assume**
  - A_1 and A_2 are sets
  - T_1/T_2 is the task of choosing an element from A_1/A_2 (a different chosen element is a different way)
- **There are $|A_1|$ ways to do T_1 and $|A_2|$ ways to do T_2**
- **The number of ways to do either T_1 or T_2**
  - $| A_1 \cup A_2| = |A_1| + |A_2| - | A_1 \cap A_2|$

- **The principle of inclusion-exclusion can be generalized to find the number of ways to do one of *n* different tasks**

Tree Diagrams

- **Some counting problems can be solved by using tree diagrams**
- **How many bit strings of length 4 do not have 2 consecutive 1s?**



Counting by Tree Diagrams

- **Best of Five (Bo5): 兩隊比賽 五戰三勝 共有幾種結局?**

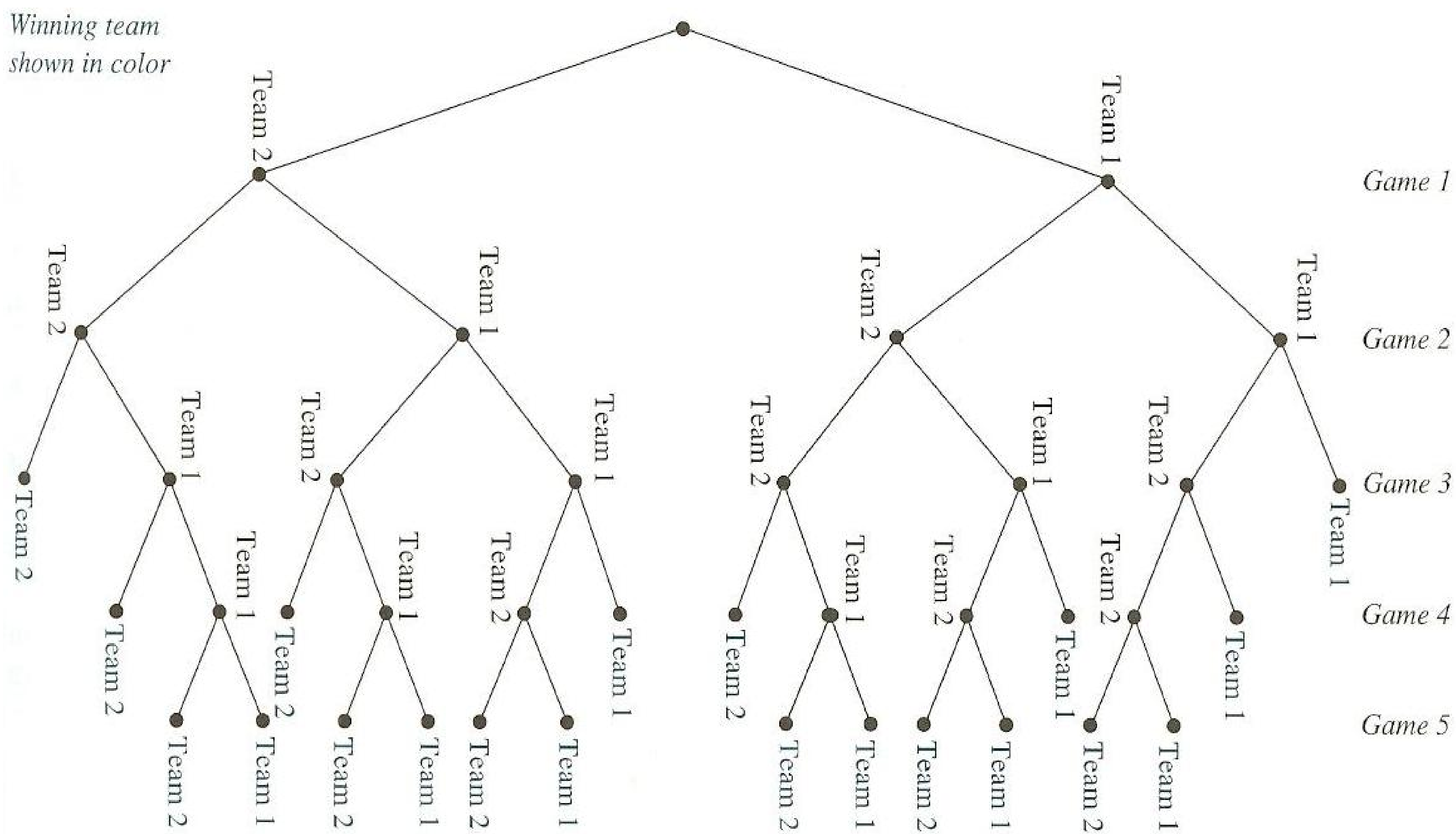
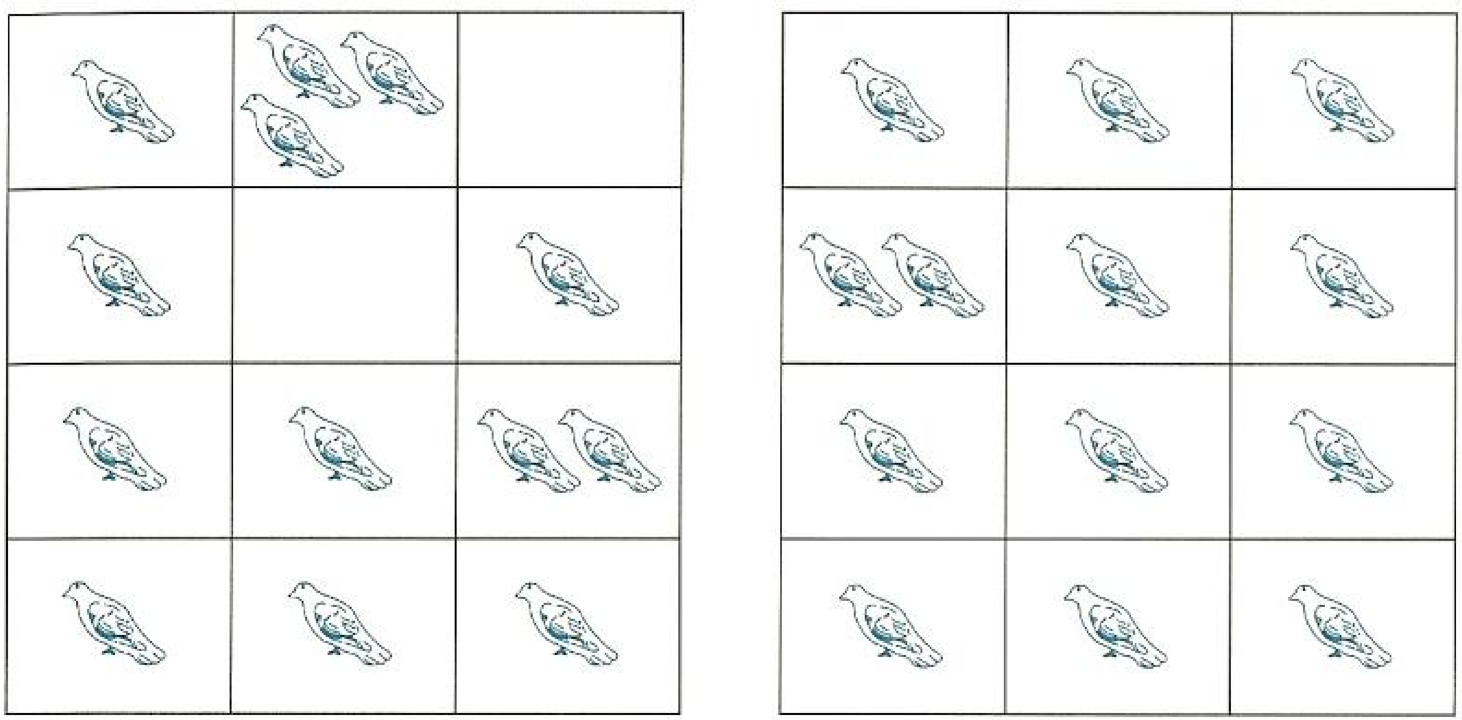


FIGURE 3 **Best Three Games Out of Five Playoffs.**

The Pigeonhole Principle

- **The pigeonhole principle**
  - If *k*+1 or more objects are placed into *k* boxes, then there is at least one box containing 2 or more of the objects.



Examples

- **In any group of 367 people, there must be at least 2 people with the same birthday**

- **In any group of 27 English words, there must be at least 2 that begins with the same initial letter**

- **How many students must be in a class to guarantee that at least 2 students get the same score on the final exam if the grade is from 0~100?**

Advanced Example

- **DIY: Show that for every positive integer *n* there is a multiple of *n* that has only 0s and 1s in its decimal expansion**
  - Consider the *n*+1 integers:
    1, 11, 111, …, $\underbrace{111\ldots11}_{n+1 \text{ 1s}}$
  - There are only *n* possible remainders when an integer is divided by *n*

Generalized Pigeonhole Principle

- **Example**
  - Among any set of 21 digits (0~9 numbers), there must be at least 3 that are the same

- **The generalized pigeonhole principle**
  - If N objects are placed into k boxes, then there is at least one box containing at least $\lceil N/k \rceil$ of the objects
  - Pf: By contradiction!
  - Suppose none of boxes contains more than $\lceil N/k \rceil$ - 1 objects.
  - Then, the total number of objects is at most
    $k * (\lceil N/k \rceil - 1) < k * ((N/k + 1) - 1) = N \rightarrow\leftarrow$

Examples

- **Among 100 people there are at least $\lceil 100/12 \rceil = 9$ people were born in the same month**
- **What is the minimum number of students in a class that at least 6 will receive the same grade? (A, B, C, D, F)**
- **DIY: How many cards must be selected from a standard deck of 52 cards to guarantee that at least 3 cards of the same suit are chosen?**
- **DIY: How many must be selected to guarantee that at least 3 hearts are selected?**

Advanced Examples (1/4)

- **Example**
  - suppose there are 15 PCs and 10 servers
  - a cable can be used to connect a PC to a server
  - What is the minimum number of cables to guarantee any set of 10 PCs can access different servers?

Advanced Examples (2/4)

- **Example**
  - During a month with 30 days
  - A baseball team plays at least 1 game a day, but no more than 45 games in this month
  - Show that there must be a period of some number of consecutive days during which the team must play exactly 14 games
- **Pf:**
  - let a_j be the number of games played on and before the jth day
  - Hence, a_1, a_2, …, a_{30} is an **increasing** sequence of **distinct** integers with $1 \leq a_j \leq 45$
  - a_1+14, a_2+14, …, $a_{30}+14$ is also an **increasing** sequence of **distinct** integers with $15 \leq a_k \leq 59$
  - These 60 positive integers are $\leq 59 \Rightarrow$ 2 integers must be identical
  - some a_k must be equal to a_j+14

Advanced Examples (3/4)

- **Show that among any n+1 positive integers not exceeding 2n, there must be an integer that divides one of the other integers**
  - n+1 positive integers: $a_1, a_2, \ldots, a_{n+1}$
  - $a_i = 2^{ki} * q_i$, where k_i is a nonnegative integer and q_i is odd
  - Since there are only n odd integers not exceeding 2n, at least 2 of the $q_1, q_2, \ldots, q_{n+1}$ must be equal
  - Assume q_m and q_n are equal, then a_m divides a_n or a_n divides a_m

Advanced Examples (4/4)

- **DIY: Every sequence of n^2 + 1 distinct real numbers containing a subsequence of length n+1 that is either strictly increasing or strictly decreasing**
  - e.g., 8, 11, 9, 1, 4, 6, 12, 10, 5, 7; 10 elements
    - strictly increasing: 1, 4, 6, 12
    - strictly decreasing: 11, 9, 6, 5
  - Pf: By contradiction
  - n^2+1 real: $a_1, a_2, \ldots, a_{n2+1}$
  - (i_k, d_k) for an a_k: i_k (or d_k) is the length of the longest increasing (or decreasing) sequence starting at a_k
  - Suppose there are no increasing/decreasing subsequences of length n+1, i.e., at most length n
  - $1 \leq i_k, d_k \leq n \Rightarrow$ only n^2 (i_k, n_k) pairs
  - By the pigeonhole principle, there exists $s<t$ with the same (i, d) pair, if $a_s < a_t$, $i_s = i_t \Rightarrow i_s \geq i_t + 1$ (include a_s) $\rightarrow\leftarrow$
  - Similarly, $a_s > a_t$ leads to a contradiction, too.