6.1 The Basics of Counting

Credit
Ming-Hsuan Yang, Richard Scherl
Mehrdad Nojoumian, Marc Pomplun,
Paul Artola, Asim Altwijri,
Husni Al-Muhtaseb

Basics of Counting

- Introduction
  - Complexity of algorithms
  - Probabilities of events
  - Other applications

6.1 Basics of Counting

- **Combinatorics**: the study of arrangements of objects
  - **Enumeration**: the counting of objects with certain properties (an important part of combinatorics)
    - a key role in mathematical biology, especially in sequencing DNA.
    - Enumerate the different telephone numbers possible in KSA
    - The allowable password on a computer
    - The different orders in which runners in a race can reach

Example

- Suppose a password on a system consists of 6, 7, or 8 characters -
- Each of these characters must be a digit or a letter of the alphabet 0, 1, ..., 9 A, B, ..., Z, a, b, ..., z
- Each password must contain at least one digit
- How many passwords are there?

Basic counting principles

- **Product rule**: Suppose that a procedure can be broken down into a sequence of two tasks. If there are n_1 ways to do the first task **and** n_2 ways to do the second task. Then there are $n_1 \cdot n_2$ ways to do the procedure.
- **Sum rule**: if a task can be done either in one of n_1 ways **or** in one of n_2 ways, where **none** of the set of n_1 ways is the same as any of the n_2 ways, then there are $n_1 + n_2$ ways to do the task.
- **Subtraction rule (inclusion-exclusion):** if a task can be done either in one of n_1 ways **or** in one of n_2 ways, then the total number of ways to do the task is $n_1 + n_2$ minus the number of ways to do the task that are common to the two different ways.
- **Division Rule**: there are n/d ways to do a task if it can be done using a procedure that can be carried out in n ways, and for every way w, exactly d of the n ways correspond to way w.

Basic counting principles

- Two basic counting principles
  - Product rule
  - Sum rule
- **Product rule**: suppose that a procedure can be broken down into a sequence of two tasks
- If there are n_1 ways to do the 1st task, and for each of these there are n_2 ways to do the 2nd task, then there are $n_1 \cdot n_2$ ways to do the procedure

Example

- The chairs of a room to be labeled with a capital letter and a positive integer not exceeding 100. What is the largest number of chairs that can be labeled differently?

A, B, ..., Z   1, 2, ..., 100

- Examples of labels using this scheme are A29, D100, C1, etc.
- There are 26 letters to assign for the 1st part and 100 possible integers to assign for the 2nd part, so there are $26 \times 100 = 2600$ different ways to label chairs

A	1
A	2
A	...
A	99
A	100
B	1
B	2
...	...
Z	1
Z	2
Z	...
Z	99
Z	100

Product rule (General)

- Suppose that a procedure is carried out by performing the tasks T_1, T_2, ..., T_m in sequence. If each task T_i, $i = 1, 2, \ldots, n$ can be done in n_i ways, regardless of how the previous tasks were done, then there are $n_1 \cdot n_2 \cdot \ldots \cdot n_m$ ways to carry out the procedure

Example

- How many different license plates are available if each plate contains a sequence of 3 capital letters followed by 3 digits (and no sequences of letters are prohibited, even if they are obscene)?
- License plate _ _ _ _ _ _ : There are 26 choices for each letter and 10 choices for each digit.

A, B, ..., Z

0, 1, ..., 9

- So, there are $26 \times 26 \times 26 \times 10 \times 10 \times 10 = 17{,}576{,}000$ possible license plates

Counting functions

- How many functions are there from a set with m elements to a set with n elements?
- A function corresponds to a choice of one of the n elements in the codomain for each of the m elements in the domain
- That is, for each of the m elements of the domain there are n choices of elements of the codomain
- Hence, by product rule there are $n \cdot n \cdot \ldots \cdot n = n^m$ functions from a set with m elements to one with n elements
- For example, there are $5^3 = 125$ different functions from a set with three elements to a set with five elements.

Counting one-to-one functions

- How many one-to-one functions are there from a set with m elements to one with n elements?
- First note that when $m > n$ there are no one-to-one functions from a set with m elements to one with n elements
- Let $m \leq n$. Suppose the elements in the domain are $a_1, a_2, \ldots, a_m$. There are n ways to choose the value for the value at a_1
- As the function is one-to-one, the value of the function at a_2 can be picked in $n - 1$ ways (the value used for a_1 cannot be used again)
- Using the product rule, there are $n\,(n - 1)\,(n - 2)\,\ldots\,(n - m + 1)$ one-to-one functions from a set with m elements to one with n elements

Example

- From a set with 3 elements to one with 5 elements, there are $5 \times 4 \times 3 = 60$ one-to-one functions

Example

- The format of telephone numbers in north America is specified by a numbering plan
- It consists of 10 digits, with 3-digit area code, 3-digit office code and 4-digit station code
- Each digit can take one form of
  - X: 0, 1, ..., 9
  - N: 2, 3, ..., 9
  - Y: 0, 1

NYX-NNX-XXXX

NXX-NXX-XXXX

Example

- In the old plan, the formats for area code, office code, and station code were NYX, NNX, and XXXX, respectively
- So the phone numbers had NYX-NNX-XXXX
- NYX: $8 \times 2 \times 10 = 160$ area codes
- NNX: $8 \times 8 \times 10 = 640$ office codes
- XXXX:$10 \times 10 \times 10 \times 10 = 10{,}000$ station codes
- So, there are $160 \times 640 \times 10{,}000 \ = \ 1{,}024{,}000{,}000$ phone numbers

X: 0, 1, ..., 9 (10 Digits)
N: 2, 3, ..., 9 (8 Digits)
Y: 0, 1 (2 Digits)

Example

- In the new plan, the formats for area code, office code, and station code are NXX, NXX, and XXXX, respectively
- So the phone numbers had NXX-NXX-XXXX
- NXX: $8 \times 10 \times 10 = 800$ area codes
- NXX: $8 \times 10 \times 10 = 800$ office codes
- XXXX:$10 \times 10 \times 10 \times 10 = 10{,}000$ station codes
- So, there are $800 \times 800 \times 10{,}000 = 6{,}400{,}000{,}000$ phone numbers

X: 0, 1, ..., 9

N: 2, 3, ..., 9

Y: 0, 1

Example

What is the value of k after the execution of the following code? where $n_1, n_2, \ldots, n_m$ are positive integers

```
k := 0
for j_1 := 1 to n_1
 for j_2 := 1 to n_2
     .
     .
     .
   for j_m := 1 to n_m
     k := k + 1
```

$n_1 n_2 \ldots n_m$

Product rule

- 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

$$|A_1 \times A_2 \times \ldots \times A_m| = |A_1| \cdot |A_2| \cdot \ldots \cdot |A_m|$$

Sum rule

- If a task can be done either in one of n_1 ways or in one of n_2 ways, where 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
- Example: suppose either a member of faculty or a student in ICS Department is chosen as a representative to a university committee. How many different choices are there for this representative if there are 28 members in faculty and 200 students?
- There are 28 + 200 = 228 ways to pick this member

Sum rule

- If $A_1, A_2, \ldots, A_m$ are disjoint finite sets, then the number of elements in the union of these sets is as follows

$$|A_1 \cup A_2 \cup \ldots \cup A_m| = |A_1| + |A_2| + \ldots + |A_m|$$

Examples

- What is the value of k after the execution of the following code? where n_1, n_2, …, n_m are positive integers

```
k := 0
for i₁ := 1 to n₁
   k := k + 1
for i₂ := 1 to n₂
   k := k + 1
   .
   .
   .
for iₘ := 1 to nₘ
   k := k + 1
```

$n_1 + n_2 + \ldots + n_m$

Inclusion-exclusion principle

- Suppose that a task can be done in n_1 or in n_2 ways, but some of the set of n_1 ways to do the task are the same as some of the n_2 ways to do the task
- Cannot simply add n_1 and n_2, but need to subtract the number of ways to the task that is common in both sets
- This technique is called **principle of inclusion-exclusion** or **subtraction principle**

Example

- Each user on a computer system has a password, which is 6 to 8 characters long, where each character is an uppercase letter or a digit. Each password must contain at least one digit. How many possible passwords are there?
- Let P be the number of all possible passwords and $P = P_6 + P_7 + P_8$ where P_i is a password of i characters
- $P_6 = 36^6 - 26^6 = 1{,}867{,}866{,}560$
- $P_7 = 36^7 - 26^7 = 70{,}332{,}353{,}920$
- $P_8 = 36^8 - 26^8 = 208{,}827{,}064{,}576$
- $P = P_6 + P_7 + P_8 = 2{,}684{,}483{,}063{,}360$

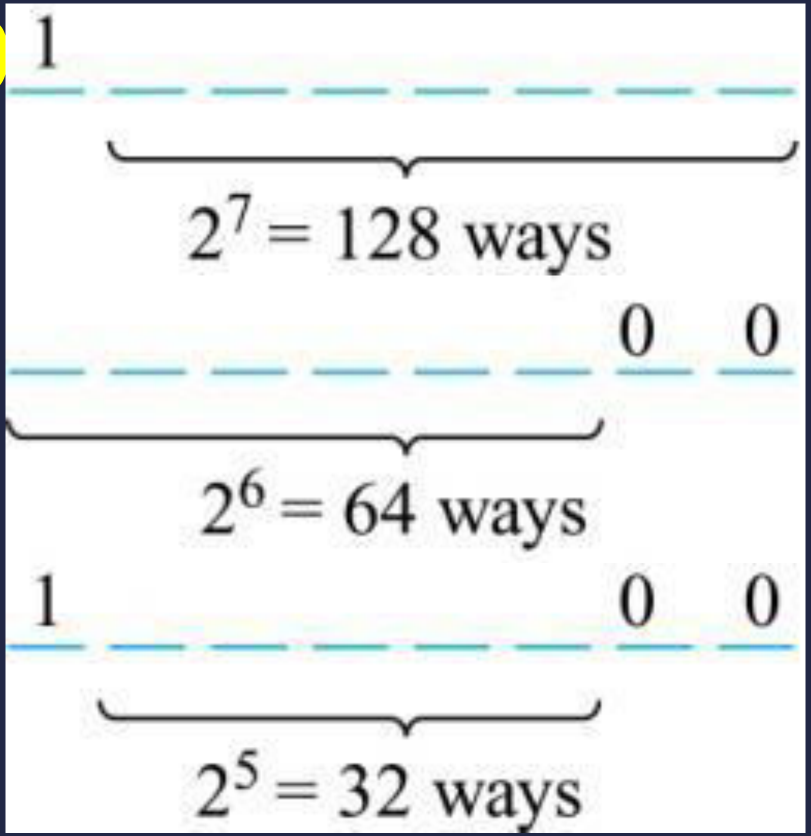
- number of strings of uppercase letters and digits that are six characters long, including those with no digits is $(26+10)^6 = 36^6$
- number of strings with no digits that are six characters long is 26^6

More Counting Problems

- In some cases, it is not enough to use only the sum rule or the product rule.
- In a version of the BASIC programming language, the name of a variable is a string of 1 or 2 alphanumeric characters, where uppercase and lowercase letters are not distinguished.
- Moreover, a variable name must begin with a letter and must be different from the five strings of two characters that are reserved for programming use
- How many different variables names are there?
- Let V_1 be the number of these variables of 1 character, and likewise V_2 for variables of 2 characters
- So, $V_1 = 26$, and $V_2 = 26 \times 36 - 5 = 931$
- In total, there are $26 + 931 = 957$ different variables

Example

- How many bit strings of length 8 either start with a 1 or end with two bits 00



1 _ _ _ _ _ _ _: $2^7 = 128$ ways

_ _ _ _ _ _ 0 0: $2^6 = 64$ ways

1 _ _ _ _ _ 0 0: $2^5 = 32$ ways

Number of possible bit strings is

128 + 64 - 32 = 160

Inclusion-exclusion principle

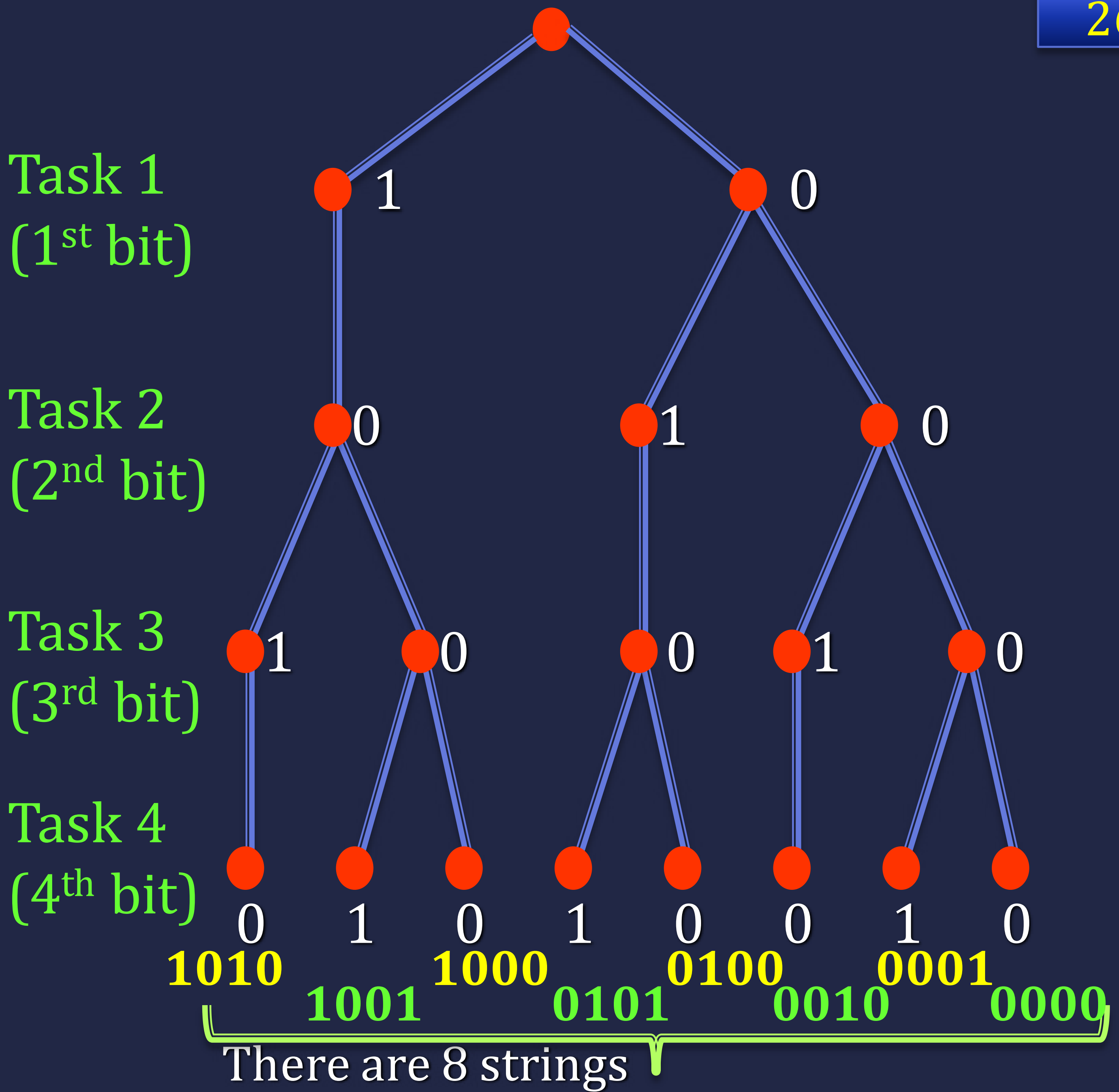
- Using sets to explain the principle

$$|A_1 \cup A_2| = |A_1| + |A_2| - |A_1 \cap A_2|$$

Tree Diagrams

In some cases, we can use tree diagrams for counting

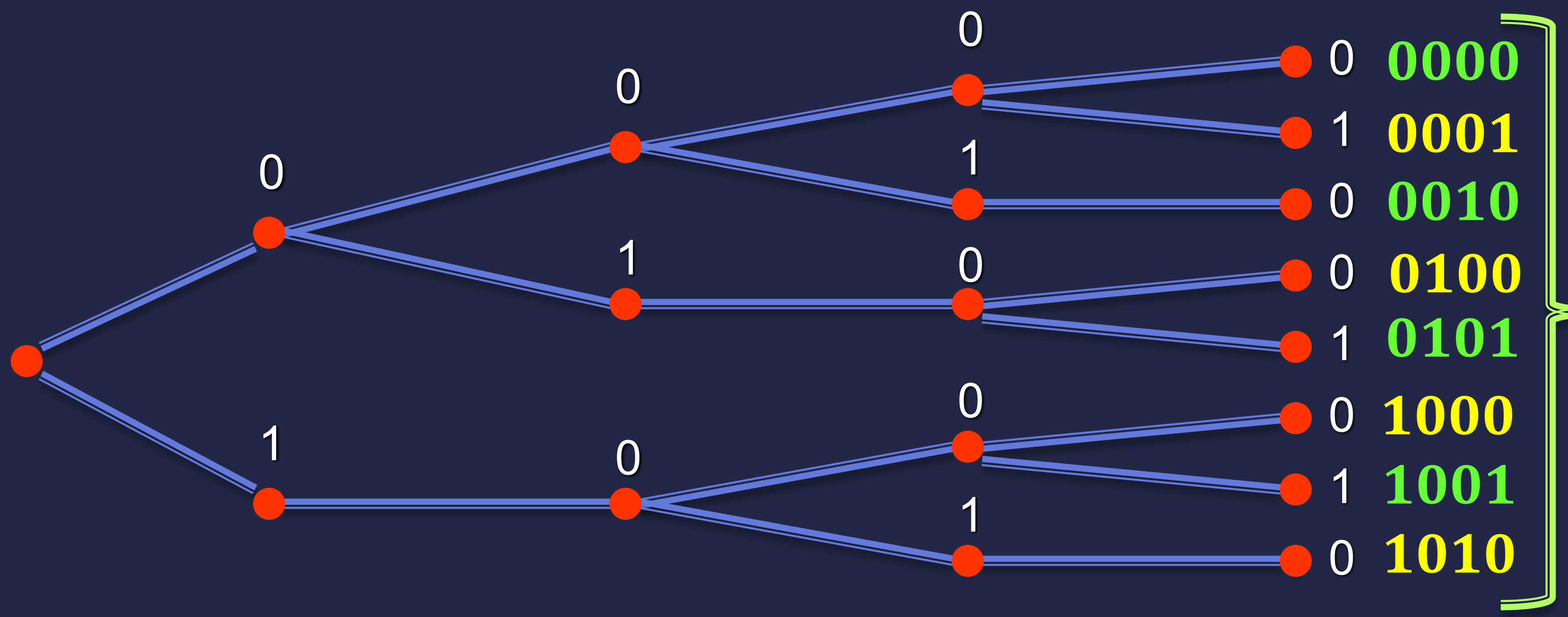
How many bit strings of length four do not have two consecutive 1s?



Sometimes you may want to have the tree from left to right

How many bit strings of length four do not have two consecutive 1s?

Task 1 (1st bit) | Task 2 (2nd bit) | Task 3 (3rd bit) | Task 4 (4th bit)



There are 8 strings

Example

- A playoff between 2 teams consists of at most 5 games. The 1st team that wins 3 games wins the playoff. How many different ways are there?

A playoff between 2 teams consists of at most 5 games. The 1st team that wins 3 games wins the playoff. How many different ways are there?

20 different ways (Terminals)

Example

- Suppose a T-shirt comes in 5 different sizes: S, M, L, XL, and XXL. Further suppose that each size comes in 4 colors, white, green, red, and black except for XL which comes only in red, green and black, and XXL which comes only in green and black. How many possible size and color of the T-shirt?

Suppose a T-shirt comes in 5 different sizes: S, M, L, XL, and XXL. Further suppose that each size comes in 4 colors, white, green, red, and black except for XL which comes only in red, green and black, and XXL which comes only in green and black. How many possible size and color of the T-shirt?

17 different T-Shirt