

6 Counting

6.1 The Basics of Counting

Product and Sum Rules

Suppose two task,

m number of ways to do task 1

n number of ways to do task 2

Do **both** task 1 and task 2:

mn ways

$$|A \times B| = |A| \cdot |B|.$$

Product Rule

Do **either** task 1 **or** task 2, but not both:

$m+n$ ways

$$|A \cup B| = |A| + |B|, \text{ if } A \cap B = \emptyset.$$

Sum Rule

Principle of Inclusion-Exclusion

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

6.2 The Pigeonhole Principle

Theorem 1 *The pigeonhole principle*

If $k + 1$ or more objects are placed into k box,

there is at least one or more box containing two or more objects.

Dirichlet drawer principle

Corollary 1 *If $f: A \rightarrow B$ and $|A| \geq |B| + 1$, then f is **not** one-to-one.*

*If $f: A \rightarrow B$ and $|A| > |B|$, then f is **not** one-to-one.*

Theorem 2 *The generalized pigeonhole principle*

If N objects are placed into k box,

there is at least one or more box containing $\lceil N/k \rceil$ objects.

proof by contradiction, *no box contains $\lceil N/k \rceil$ objects.*

$$k(\lceil N/k \rceil - 1) < k((N/k + 1) - 1) = N$$

Example $N = 59$ students, $k = 12$ months,

$$\lceil N/k \rceil = \lceil 4.91\dots \rceil = 5 \text{ students/month}$$

6.3 Permutations and Combinations

Thm. 1 For $n \geq 0$ and $1 \leq \forall r \leq n$, $P(n, r) = n(n-1) \dots (n-r-1)$

Fact. 1 $P(n, 0) = n$.

Col. 1 For $n \geq 0$ and $0 \leq \forall r \leq n$, $P(n, r) = n! / (n-r)!$

Thm. 2 $0 \leq r \leq n$, $C(n, r) = P(n, r) / P(r, r) = n! / r!(n-r)!$

Col. 2 $C(n, r) = C(n, n-r)$

6.4 Binomial Coefficient

Thm. 1 $(x+y)^n = \sum_{j=0}^n C(n, j) \cdot x^{n-j} \cdot y^j$
 $= C(n, 0) \cdot x^n \cdot y^0 + C(n, 1) \cdot x^{n-1} \cdot y^1 + \dots + C(n, n-k) \cdot x^{n-k} \cdot y^k + \dots +$
 $C(n, n-1) \cdot x^1 \cdot y^{n-1} + C(n, n) \cdot x^0 \cdot y^n. \quad \text{for } 1 \leq \forall k \leq n.$

proof $C(n, n-k)$ terms of $x^{n-k} \cdot y^k$ among 2^n expanded terms of $(x+y)^n$.

Col. 1 $\sum_{k=0}^n C(n, k) = 2^n$ for n .

proof $2^n = (1+1)^n = \sum_{k=0}^n C(n, k) 1^k 1^{n-k} = \sum_{k=0}^n C(n, k).$

Col. 2 $\sum_{k=0}^n (-1)^k C(n, k) = 0.$

$0 = 0^n = (1+(-1))^n = \sum_{k=0}^n C(n, k) (-1)^k 1^{n-k} = \sum_{k=0}^n (-1)^k C(n, k).$

Col. 3 $\sum_{k=0}^n 2^k C(n, k) = 3^n.$

$3^n = (1+2)^n = \sum_{k=0}^n C(n, k) 1^{n-k} 2^k = \sum_{k=0}^n C(n, k) 2^k.$

Pascal's Identity and Triangle

Thm. 2 $C(n+1, k) = C(n, k-1) + C(n, k)$

Proof $C(n, k-1) + C(n, k) = n! / [(n-k+1)! \cdot (k-1)!] + n! / [(n-k)! \cdot k!]$
 $= [n! \cdot k + n! \cdot (n-k+1)] / [(n-k+1)! \cdot k!]$
 $= n!(n+1) / [(n-k+1)! \cdot k!] = (n+1)! / [(n-k+1)! \cdot k!] = C(n+1, k)$

Pascal's triangle

$$\begin{array}{ccccccc}
 & & & & C(0, 0) & & \\
 & & & & C(1, 0) & C(1, 1) & \\
 & & & & \dots & & \\
 & & & & C(n, 0) & \dots & C(n, k-1) & C(n, k) & \dots & C(n, n) \\
 C(n+1, 0) & \dots & C(n+1, k-1) & C(n+1, k) & C(n+1, k+1) & \dots & C(n, n)
 \end{array}$$

6.5 Generalized Permutation and Combinations

Permutation with Repetition

Thm 1 $\Pi(n, r) = n \cdot n \cdot \dots \cdot n = n^r$.

$$P(n, r) = n! / (n-r)!$$

Combination with Reprtition

Thm 2 $H(n, r) = C(n + r - 1, r) = C(n + r - 1, n - 1)$

$$= (n+r-1)! / r!(n-1)!$$

$$C(n, r) = n! / r!(n-r)!$$

6.6 Generating Permutation and Combinations

Let $a_1a_2\dots a_n$ and $b_1b_2\dots b_n$ be permutation of $\{1, 2, \dots, n\}$.

We say $a_1a_2\dots a_n < b_1b_2\dots b_n$,

if $1 \leq \exists k \leq n \ .\exists . a_1 = b_1, a_2 = b_2, \dots, a_{k-1} = b_{k-1}, a_k < b_k$.

permutation $a_1a_2\dots a_n \in \{1, \dots, n\}^n$.

$<$ is a total order

Next permutation of $a_1 \dots a_n$?

Find largest j $\exists. a_j < a_{j+1}$.

Find smallest interger a_k in a_{j+1}, \dots, a_n but $a_k > a_j$.

Interchange a_k and a_j .

Increasing order from $j+1$ -th to n -th position.

procedure *nextPermutation*($a_1 \dots a_n \in \{1, \dots, n\}^n \neq n (n-1) \dots 2 1$)

j := n-1; do $a_j > a_{j+1} \rightarrow j := j - 1$ ***od***

k := n; do $a_j > a_k \rightarrow k := k - 1$ ***od***

$a_k, a_j := a_j, a_k$;

r, s := n, j+1; do $r > s \rightarrow a_r, a_s := a_s, a_r$; ***r, s := r - 1, s+1 od***

procedure *next bit string*($b_1 \dots b_n \in \{0, 1\}^n \neq 111 \dots 11$)
 $i := n$; **do** $b_i = 1 \rightarrow b_i := 0$; $i := i - 1$ **od**; $b_i := 1$.

Next r -combination of $\{a_1, a_2, \dots, a_n\}$ with $a_1 < a_2 < \dots < a_n$?

Find $a_i \exists. a_i \neq n - r + i$.

replace a_i with $a_i + 1$.

repalace a_j 's with $a_i + j - i$.

procedure *next r-comnination*($\{a_1, a_2, \dots, a_n\} \subseteq \{1, 2, \dots, n\}$
 $\neq \{n-r+1, n-r+2, \dots, n\}$ and $a_1 < a_2 < \dots < a_n$)
 $i := r$; **do** $a_i = n - r + i \rightarrow i := i - 1$ **od**; $a_i := a_i + 1$;
for $j := i + 1$ **to** r **do** $a_j := a_i + j - i$ **od**