# Review VII(Slides 353 - 416)

**Counting! Counting!**

Counting something is never easy

HamHam

University of Michigan-Shanghai Jiao Tong University Joint Institute

February 9, 2022

VE203 - Discrete Mathmatics

# Twelvefold Way (will be provided)

Distribute $k$ balls into $n$ urns ($f\colon B \to U$, $|B| = k$, $|U| = n$).

| Balls (domain) | Urns (codomain) | unrestricted (any function) | $\leq 1$ (injective) | $\geq 1$ (surjective) |
|---|---|---|---|---|
| labeled | labeled | $n^k$ | $n^{\underline{k}}$ | $n! \left\{ {k \atop n} \right\}$ |
| unlabeled | labeled | $\left(\!\binom{n}{k}\!\right)$ | $\binom{n}{k}$ | $\left(\!\binom{n}{k-n}\!\right)$ |
| labeled | unlabeled | $\sum_{i=1}^{n} \left\{ {k \atop i} \right\}$ | $[k \leqslant n]$ | $\left\{ {k \atop n} \right\}$ |
| unlabeled | unlabeled | $\sum_{i=1}^{n} p_i(k)$ | $[k \leqslant n]$ | $p_n(k)$ |

# A few names...

Notations in red are suggested.

- Permutation:

$$n^{\underline{k}} = P(n,k) = P_k^n = \underbrace{n \cdot (n-1) \cdot (n-2) \cdots (n-k+1)}_{k \text{ terms}} = \frac{n!}{(n-k)!},$$

$$(A_n^k \text{ in middle school, } n \geqslant k)$$

- Combination: $\binom{n}{k} = C(n,k) = C_k^n = \dfrac{n!}{k!\,(n-k)!},\ (n \geqslant k)$
- Multichoosing: $\left(\!\binom{n}{k}\!\right) = \binom{n+k-1}{k} \quad (n < k \text{ is possible!})$
- Circular permutation: $\dfrac{P(n,k)}{k!}$

## Basic Properties

- $\binom{n}{k} = \binom{n}{n-k} = \frac{n}{k}\binom{n-1}{k-1}$
- $\binom{n}{0} = \binom{n}{n} = 1$, $\binom{n}{1} = n$
- $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$
- $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$, (binomial theorem)
  - Commonly used form: $(x+1)^n = \sum_{k=0}^{n} \binom{n}{k} x^k$
  - Application: $\sum_{k=0}^{n} \binom{n}{k} = 2^n$, $\sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0$
- $\left\{ {n \atop k} \right\} = \left\{ {n-1 \atop k-1} \right\} + k \left\{ {n-1 \atop k} \right\}$

# Exercise

1. Prove the following equations:

   1. $$\sum_{k=0}^{n} k\binom{n}{k} = n \cdot 2^{n-1}$$

   2. $$\sum_{k=0}^{n} k^2\binom{n}{k} = n(n+1) \cdot 2^{n-2}$$

# Exercise

We can have something related to modular arithmetic!

2. If $p$ is a prime, prove the following:
   - $\binom{n}{p} \equiv \lfloor \frac{n}{p} \rfloor \pmod{p}$
   - $\binom{p}{k} \equiv 0 \pmod{p}$, for $1 \leq k \leq p-1$
   - $\binom{p-1}{k} \equiv (-1)^k \pmod{p}$, for $0 \leq k \leq p-1$
   - $\binom{p+1}{k} \equiv 0 \pmod{p}$, for $2 \leq k \leq p-1$

# Multichoosing

## Definition

Multiset: Let $S$ be a multiset,

$$S = \{n_1 * a_1, n_2 * a_2, \cdots, n_k * a_k\}, \ n = n_1 + n_2 + \cdots + n_k,$$

which means there are $n_1$ element $a_1$, $n_2$ element $a_2$, $\cdots$, $n_k$ element $a_k$. $n_i$ is multiplicity of $a_i \in S$.

Permutation in multiset:

- $P(n, n) = \dfrac{n!}{n_1!n_2!\cdots n_k!}$
  $\rightarrow$Proof: select by steps

$$P(n, n) = \binom{n}{n_1}\binom{n-n_1}{n_2}\cdots\binom{n-n_1-n_2-\cdots-n_{k-1}}{n_k}.$$

# Multichoosing

**Combination in multiset:**

- $\left(\!\binom{k}{r}\!\right) = \binom{k+r-1}{r}$ if $r \leqslant \forall n_i$.

$\rightarrow$Proof:

The combination $\left(\!\binom{k}{r}\!\right)$ equals to the number of solutions to the indefinite equation

$$x_1 + x_2 + \cdots + x_k = r.$$

A solution corresponds to a permutation

$$\underbrace{1\cdots1}_{x_1 \text{ times}}\,0\,\underbrace{1\cdots1}_{x_2 \text{ times}}\,0\cdots0\,\underbrace{1\cdots1}_{x_k \text{ times}}. \tag{1}$$

*I.e.*, if $k-1$ 0s divide $r$ 1s into $k$, $x_i$ is the number per partition.

## Properties

- $\left(\!\!\binom{k}{r}\!\!\right) = \left(\!\!\binom{r+1}{k-1}\!\!\right)$

  $\rightarrow$Proof: number of ways to arrange $k-1$ 0s and $r$ 1s = number of ways to arrange $r$ 1s and $k-1$ 0s.

$$LHS = \left(\!\!\binom{(k-1)+1}{r}\!\!\right) = \left(\!\!\binom{r+1}{k-1}\!\!\right) = RHS.$$

- Multinomial formula:

$$(a_1 + \cdots + a_m)^n = \sum_{\substack{k_1,\ldots,k_m \geqslant 0 \\ k_1+\cdots+k_m=n}} \binom{n}{k_1, k_2, \ldots, k_m} a_1^{k_1} a_2^{k_2} \cdots a_m^{k_m}$$

# Exercise

Of course, we can have something more fancy. But...the exam actually prefers the exercise below 😅😅😅

3. Find the number of **non-negative** integers solutions of

$$x_1 + x_2 + x_3 + x_4 = 30,$$

such that $3 \leqslant x_i \leqslant 10$ for every $1 \leqslant i \leqslant 4$.

# Exercise

Of course, we can have something more fancy. But...the exam actually prefers the exercise below 😅 😅 😅

3. Find the number of **non-negative** integers solutions of

$$x_1 + x_2 + x_3 + x_4 = 30,$$

such that $3 \leqslant x_i \leqslant 10$ for every $1 \leqslant i \leqslant 4$.

Answer:

- $\left(\!\binom{5}{10}\!\right) = \binom{14}{10} = 1001.$
- $\left(\binom{14}{10} - \left(\binom{12}{2} + \binom{10}{2} + \cdots + \binom{2}{2}\right)\right)/2 = 420.$

# Inclusion-Exclusion Principle

## Notation

Given $I \subset \{1, \cdots, n\}$, let

$$A_I = \bigcap_{i \in I} A_i,$$

where $A_i \subset X$ for all $i \in I$.

For example, $A_{\{1,2,4\}} = A_1 \cap A_2 \cap A_4$. In particular, $A_\varnothing = X$.

## Theorem (Inclusion-Exclusion Principle)

Let $A_1, \cdots, A_n$ be subsets of $X$. Then the number of elements of $X$ which lie in none of the subsets $A_i$ is

$$\sum_{I \subset \{1,\ldots,n\}} (-1)^{|I|} |A_I| .$$

## Corollary

Let $A_1, \cdots, A_n$ be a sequence of (not necessarily distinct) sets, then

$$|A_1 \cup A_2 \cup \cdots \cup A_n| = \sum_{\varnothing \neq I \subset \{1,\ldots,n\}} (-1)^{|I|+1} |A_I| .$$

### Special Case

When $|I| = |J| \Rightarrow |A_I| = |A_J|$

$$|A_1 \cup A_2 \cup \cdots \cup A_n| = \sum_{|I|=1} (-1)^{|I|+1} \binom{n}{|I|} |A_I| .$$

### Derangement

$$d_n = \sum_{i=0}^{n} (-1)^i \binom{n}{i} (n-i)! = n! \sum_{i=0}^{n} \frac{(-1)^i}{i!}$$

# Exercise

4. Find the number of non-negative integers solutions of

$$x_1 + x_2 + x_3 + x_4 = 30$$

such that $3 \leqslant x_i \leqslant 10$ for every $1 \leqslant i \leqslant 4$.

Answer: $1330 - 1084 = 246$.

# Counting Surjections

Let $k \geqslant n$. The number of surjections $f : \{1, \cdots, k\} \to \{1, \cdots, n\}$ is given by

$$S_{k,n} = \sum_{i=0}^{n-1} (-1)^i \binom{n}{i} (n-i)^k = n! \left\{ \begin{matrix} k \\ n \end{matrix} \right\}.$$

## Exercise

Which of the following identities are valid for $n = 2021$?

(A) $\sum_{k=0}^{n} \binom{n}{k} (n-k)^{n-2} (-1)^k = 0$

(B) $\sum_{k=0}^{n} \binom{n}{k} (n-k)^{n-1} (-1)^k = n!$

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

(D) $\sum_{k=0}^{n} \binom{n}{k} (n-k)^{n+1} (-1)^k = n \cdot (n+1)!/2$

# Matrix Chain Multiplication

This is the boring recipe...

1. we shall create a 2-dimensional array for storing $m[i, j]$ costs needed to compute $A_{i \cdots j}$.
2. Remember that if we have only one matrix in a sequence, then there is nothing to multiply. It means that $m[i, j] = 0$, when $i = j$.
3. However, when $i < j$, we split the sequence into $A_{i \cdots k}$ and $A_{k+1 \cdots j}$. Then the total cost for $A_{i \cdots j}$ will be based on the following recursive rule (DP Equation):

$$m[i, j] = \min_{i \leqslant k < j} \{m[i, j] = m[i, k] + m[k+1, j] + p_{i-1} p_k p_j\}$$

4. We cannot have $i > j$, thus the part of the table below the main diagonal will be ignored.

# Exercise

If this does occur in the exam, it would be like this one.

5. Find an optimal parenthesization of a matrix chain multiplication whose sequence of dimension is $(20, 2, 30, 12, 8)$.

Answer:

The optimal parenthesization is $A_1((A_2A_3)A_4)$.

# Linear Recurrence Relation

Forgive me that it is really difficult to summerize a **general recipe**. Let's just look at a concrete example. I think it contains almost everything.

**Example**

Find the general solution to the following inhomogeneous linear recurrence equation

$$a_{n+3} = 8a_{n+2} - 21a_{n+1} + 18a_n + 5^n + (n^2 + 1)3^n \tag{2}$$

## Solution

Like solving differential equations, the general solution is given by

$$a_n = a_n^{part} + a_n^{hom}$$

For the homogeneous part we establish the characteristic equation:

$$A^3 = 8A^2 - 21A + 18$$

Press CASIO or by factorization:

$$(A-2)\cdot(A-3)^2 = 0$$

Hence, the homogeneous solution is given by:

$$a_n^{hom} = c_1 2^n + c_2 3^n + c_3 n \cdot 3^n$$

# Solution (Cont.)

For the first inhomogeneous part $5^n$, it is easy to see

$$a_n^{part_1} = c_4 5^n$$

Plug in back to Eq. 2,

$$c_4 = 8c_4 - 21c_4 + 18c_4 + 1 \Rightarrow c_4 = -\frac{1}{4}$$

For the second inhomogeneous part $(n^2+1)3^n$, the root $A = 3$ has multiplicity of 2 and $n^2 + 1$ is a polynomial of **degree of 2**, so:

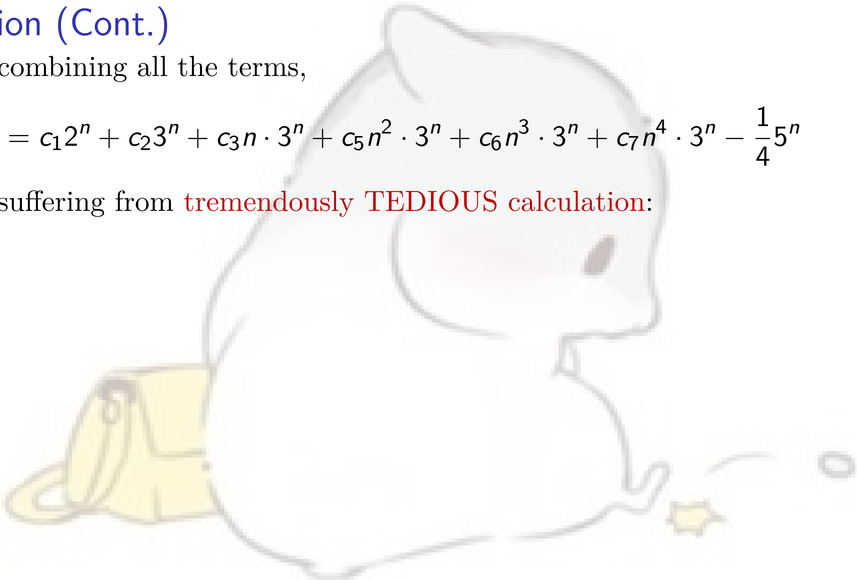
$$a_n^{part_2} = (c_5 + c_6 n + c_7 n^2) n^2 3^n$$

## Solution (Cont.)

Then combining all the terms,

$$a_n^{gen} = c_1 2^n + c_2 3^n + c_3 n \cdot 3^n + c_5 n^2 \cdot 3^n + c_6 n^3 \cdot 3^n + c_7 n^4 \cdot 3^n - \frac{1}{4} 5^n$$

After suffering from tremendously TEDIOUS calculation:



## Solution (Cont.)

Then combining all the terms,

$$a_n^{gen} = c_1 2^n + c_2 3^n + c_3 n \cdot 3^n + c_5 n^2 \cdot 3^n + c_6 n^3 \cdot 3^n + c_7 n^4 \cdot 3^n - \frac{1}{4} 5^n$$

After suffering from tremendously TEDIOUS calculation:

$a_n^{gen} =$ 🤔🤔🤔🧐🧐🧐😖😖😖😴😴😴🤗🤗🤗😅😅😅

### Exam Question

Find the general solution to

$$y_{n+2} - 5y_{n+1} + 6y_n = n^2 \cdot 3^n$$

Answer:

$$y_n = c_1 2^n + \left( c_2 + \frac{109}{18} n - \frac{7}{6} n^2 + \frac{1}{8} n^3 \right) 3^n$$

# Catalan Numbers

## Definition

$$C_n = \frac{1}{n+1}\binom{2n}{n} \qquad n \in \mathbb{N}$$

## Recurrence Relation

$$C_n = \sum_{k=0}^{n-1} C_k C_{n-1-k}$$

Fancy Examples:

- The number of ways to parenthesize the product of $n+1$ numbers
- The number of rooted trees with $n$ edges.
- The number of stack permutation of $n$ elements.
- The number of triangulations of a polygon with $n+2$ sides.

# Explanation for Recurrence Relation

We consider the situation that adding bracket to the product $x_0 \cdot x_1 \cdot x_2 \cdot \cdot x_n$. Note that no matter how we add the bracket, there is one "$\cdot$" that is outside all the bracket. [e.g. $(x_0 \cdot (x_1 \cdot x_2)) \cdot x_3$, the last operator] We consider this operator to appear between $x_k$ and $x_{k+1}$ , there exists $C_k C_{n-k-1}$ approaches to add the brackets to determine the order of multiplication. The reason is that, there are $C_k$ ways to adding brackets to $x_0 \cdot x_1 \cdots x_k$, and $C_{n-k-1}$ ways for $x_{k+1} \cdot x_{k+2} \cdots x_n$. Since the last operator could appear between any two among these $n+1$ numbers, so that

$$
\begin{aligned}
C_n &= C_0 C_{n-1} + C_1 C_{n-2} + \cdot + C_{n-2} C_1 + C_{n-1} C_0 \\
&= \sum_{k=0}^{n-1} C_k C_{n-k-1}
\end{aligned}
$$

Note that the initial value should be $C_0 = 1$ and $C_1 = 1$.

# Generating Function

Let $G(x) = \sum_{n=0}^{\infty} C_n x^n$ be the generating function for $\{C_n\}$. Then

$$G(x)^2 = \sum_{n=0}^{\infty}\left(\sum_{k=0}^{n} C_k C_{n-k}\right)x^n$$

$$= \sum_{n=1}^{\infty}\left(\sum_{k=0}^{n-1} C_k C_{n-1-k}\right)x^{n-1}$$

$$= \sum_{n=1}^{\infty} C_n x^{n-1}$$

Hence, $xG(x)^2 = \sum_{n=1}^{\infty} C_n x^n$, which implies $xG(x)^2 - G(x) + 1 = 0$. By solving this equation we can get $G(x) = (1 \pm \sqrt{1-4x})/(2x)$ .

We choose the minus sign because the plus sign would lead to a division by zero.

# Dynamic Programming

Dynamic programming (DP) is both a mathematical optimization method and a computer programming method. The method was developed by **Richard Bellman** in the 1950s and has found applications in numerous fields.

DP actually occurs in the slides! Recall:

- Matrix Chain Multiplication
- Longest Non-decresing Subsequence
- Linear Recurrence Relation
- ...

Common Features

- same sub-problem
- optimal sub-structure

# Where did the name come from?

An interesting question is, "Where did the name, dynamic programming, come from?"

The 1950s were not good years for mathematical research. We had a very interesting gentleman in Washington named Wilson. He was Secretary of Defense, and he actually had a pathological fear and hatred of the word "research". I'm not using the term lightly; I'm using it precisely. His face would suffuse, he would turn red, and he would get violent if people used the term research in his presence. You can imagine how he felt, then, about the term mathematical.

The RAND Corporation was employed by the Air Force, and the Air Force had Wilson as its boss, essentially. Hence, I felt I had to do something to shield Wilson and the Air Force from the fact that I was really doing mathematics inside the RAND Corporation.

# Where did the name come from?

What title, what name, could I choose? In the first place I was interested in planning, in decision making, in thinking. But planning, is not a good word for various reasons. I decided therefore to use the word “programming”. I wanted to get across the idea that this was dynamic, this was multistage, this was time-varying. I thought, let’s kill two birds with one stone. Let’s take a word that has an absolutely precise meaning, namely dynamic, in the classical physical sense. It also has a very interesting property as an adjective, and that is it’s impossible to use the word dynamic in a pejorative sense. Try thinking of some combination that will possibly give it a pejorative meaning. It’s impossible. Thus, I thought dynamic programming was a good name. It was something not even a Congressman could object to. So I used it as an umbrella for my activities.

# Backpack Problem

Another typical example of DP is the Backpack Problem.

[Problem Statement]

We have There are $N$ items and a backpack of capacity $V$. The i-th item has a weight $w_i$ and a value $v_i$. Figure out which items to put into the backpack, so that the total weight of these items does not exceed the backpack capacity, and the total value reaches maximum.

[DP Equation]

$$f[v] = \max\left(f[v], f[v - v_i] + w_i\right)$$

[Example Question]

https://vijos.org/p/1104

# Sample Code

```
#include <iostream>
using namespace std;
inline int max(int a,int b){return a>b?a:b;}
int main(){
    int t,m,w,c;
    cin >> t >> m;
    int f[1001]={0};
    for (int i = 0; i < m; i++){
        cin >> w >> c;
        for (int j = t; j >= w; j--){
            f[j]=max(f[j-w]+c,f[j]);
        }
    }
    int max_value=0;
    for (int i = 0; i <= t; i++){
        max_value=max(max_value,f[i]);
    }
    cout << max_value;
    return 0;
}
```

# Reference


- Example From Horst's Slides FA2020.
- Exercises from Ve203-2021-Fall TA Zhao Jiayuan.
- Exercises from 2021-Fall-Ve203 Mid_2 Exam.
- Kenneth, H.Rosen. Translated by Xu Liutong etc. *Discrete Mathematics amd Its Applications*, Eightth Edition, Chinese Abridgement. China Machine Press, 2019 print.
- Richard Bellman, Eye of the Hurricane: An Autobiography (1984, page 159)
  `http://en.volupedia.org/wiki/Dynamic_programming`