Salute to 郭阳

目录

其它作品下载链接. (click here) Turn up the sound. (my tunes) chzh_xie@foxmail.com. (contact)

Chapter01.行列式 Determinent

在 my book1 的倒数第二个主题里，我们通过对 n×n 的线性方程组，采用第 i 次对应 n-i 对相邻方程消去末尾未知数x_{n-i+1}，共 n-1 次方块长宽-1、C_n^2次俩方程间的消元，得到了最终剩下的单个方程中的x_1的系数表达式，以及该系数各项因子及其指数的数学表达式、及其证明过程。同样，在这个过程中，我们也顺带尝试着复现了行列式是如何起源于线性方程组求解问题的，这个历史过程。

行列式本质上是个多项式(如果数组由字母填充)，或者是个数、值(如果数组由数字填充)。将行列式的符号——两根竖线“| |”这个绝对值符号一样的符号，左一根右一根地把一个二维数组(比如“$\begin{matrix}1&2&3\\4&5&6\\7&8&9\end{matrix}$”)夹在其中(如“$\begin{vmatrix}1&2&3\\4&5&6\\7&8&9\end{vmatrix}$”)，表示我们要用某种运算法则，对这个二维数组中的所有元素进行某种抽象的运算，然后映射输出为一个多项式(或者说一个格式为“多项式”的应用程序，or 一个变量类型为“多项式”的变量、一个函数的返回值类型为“多项式”的返回值)。

所以说，运算符号“| |”相当于一个函数主体，是个 entrance、接口，而往里放、往里填空的二维数组“$\begin{matrix}a_{11}a_{12}a_{13}\\a_{21}a_{22}a_{23}\\a_{31}a_{32}a_{33}\end{matrix}$”，是一堆赋了值 or 尚未赋值的变量，他们以二维数组的格式储存和排列在一起。“$\begin{matrix}a_{11}a_{12}a_{13}\\a_{21}a_{22}a_{23}\\a_{31}a_{32}a_{33}\end{matrix}$” + “| |”既可以看作是一种操作，又可以看作是操作后的结果(许多数学符号均可以这样看，比如2^{1998})。而由于这种“操作”“有点复杂”，所以我们取而代之地用抽象的数据存储方式“$\begin{matrix}a_{11}a_{12}a_{13}\\a_{21}a_{22}a_{23}\\a_{31}a_{32}a_{33}\end{matrix}$”、以及抽象的数学运算符号“| |”，来简明扼要地表示这个函数运行过程和结果：“$\begin{vmatrix}a_{11}a_{12}a_{13}\\a_{21}a_{22}a_{23}\\a_{31}a_{32}a_{33}\end{vmatrix}$”。

1.1 高阶行列式的定义

郭老提出过这么个有意思的话题：|-5|=?，老人家会说，学了线性代数你就该知道，|-5|=±5。很容易想到，这是利用了绝对值符号和行列式符号的相似之处。——也就是说，对于|-5|=-5，只有在“| |”充当“行列式符号”的含义时，才可能成立。但为什么它成立呢？——因为“行列式符号：| |”操作的对象是 2 维数组而不是 0 维数组(单变量)，即 input 的是 2 维数组，而 output 的却是 0 维数组，即 2 维→0 维。

当|-5|=5 时，该等式成立的充要条件为：“| |” = “绝对值符号”，那么由于“绝对值符号”函数所对应的映射规则为 0 维数组→0 维数组，所以作为 input 的“-5”，由于必须符合框架，而只能在含义上被认为是 0 维数组；同样，|-5|=-5 要成立⇔“| |” = “行列式符号”⇔作为行列式函数的 input 的“-5”，在含义上为“只储存了一个变量的 2 维数组”。

“只储存了一个变量的 2 维数组” + “| |” =|-5|，便是一个最简单的行列式，称为一阶行列式。现在我们开始我们的叙述 journey：

1.一阶行列式

针对一元一次方程 ax=b，当|a|=a≠0 时，有唯一解 $x=\frac{|b|}{|a|}=\frac{b}{a}$。

2.二阶行列式

对于二元线性方程组(线性=一次)，$a_{11}x_1+a_{12}x_2=b_1$ & $a_{21}x_1+a_{22}x_2=b_2$，当 $\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}=a_{11}|a_{22}|-a_{12}|a_{21}|=a_{11}a_{22}-a_{12}a_{21}\neq 0$ 时，有唯一解 $x_1=\frac{\begin{vmatrix} b_1 & a_{12} \\ b_2 & a_{22} \end{vmatrix}}{\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}}$ 和 $x_2=\frac{\begin{vmatrix} a_{11} & b_1 \\ a_{21} & b_2 \end{vmatrix}}{\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}}$。

其中形如 $\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$ 的式子，即为二阶行列式；之前我们说过，“$\begin{matrix} a_{11}a_{12} \\ a_{21}a_{22} \end{matrix}$” + “| |”

$=\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$既可以看作是一种操作，又可以看作是操作后的结果——那么对于二阶行列式而言，“这种操作”便是：$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}=a_{11}|a_{22}|-a_{12}|a_{21}|=a_{11}a_{22}-a_{12}a_{21}$；而“这种操作的结果”便是：$a_{11}a_{22}-a_{12}a_{21}$；而$a_{11}a_{22}-a_{12}a_{21}=\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$，所以说行列式的本质是个多项式喃。

需要注意的是，二阶行列式的定义甚至之后的行列式的定义，均是“位置型”的定义，即其定义中所涉及的指代名词所指的是“某个位置的变量”，着眼于“位置”，而不是就事论事地指那某个字母。比如上面一段中的“形如”二字，就在隐含地强调“位置”信息的匹配。再比如a_{12}，我们完全可以换成m_{12}，我们在意的是它的下标，或者说，该下标所对应的位置中的变量它。——因此对于二阶行列式而言，其值，或者说其展开式，或者说其行列式符号对被夹在其中的数组的操作为：行列式的主对角线(左上角↘右下角)上两个元素之积，减去次对角线(左下角↙右上角)上两个元素之积。

3.三阶行列式

若我们记系数行列式(由系数构成的行列式)$D=\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$、$D_i$为系数行列式 D 的第 i 列被[方程组的常数项所排成的列←要知道我们的方程组是竖着平行排列一个个横向方程所得的方块]置换(置换一词：旧的元素被丢弃，新的元素来占据此位置，并且不管旧元素的去向；暗示这又是一个位置型定义)后，得到的新行列式。比如，对于二阶的D_i们而言，$D_1=\begin{vmatrix} b_1 & a_{12} \\ b_2 & a_{22} \end{vmatrix}$，$D_2=\begin{vmatrix} a_{11} & b_1 \\ a_{21} & b_2 \end{vmatrix}$。

当方程组的系数行列式 D≠0 时，方程组有唯一解：$x_i=\frac{D_i}{D}$。该法则也适用于三元及以上的线性方程组，称为克莱姆 Cramer 法则。

那问题来了，三阶系数行列式 $D=\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$以什么样的规则展开呢？类似二阶的$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}=a_{11}|a_{22}|-a_{12}|a_{21}|$，我们有$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}=a_{11}\begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix}-a_{12}\begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix}+$

$a_{13}\begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}$，你还可以继续将其中的 2 阶→1 阶展开，不过为了观察规律我们就此打住。

4.n 阶行列式

从 1、2、3 阶的(系数)行列式可见一斑，将一个 n 阶行列式$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$化为→$C_n^1$ 个 [1 阶行列式与 n-1 阶行列式之积] 之和的规则如下：

$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=a_{11}A_{11}+a_{12}A_{12}+\ldots+a_{1n}A_{1n}$，其中$A_{1j}=(-1)^{1+j}M_{1j}$，称为元素$a_{1j}$的代数余子式。而$M_{1j}$=将 n 阶行列式划掉第 1 行和第 j 列的所有元素后，剩余元素间的相对位置不变$\begin{vmatrix} a_{21} & \cdots & a_{2,j-1} & & a_{2,j+1} & \cdots & a_{1n} \\ a_{31} & \cdots & a_{3,j-1} & & a_{3,j+1} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,j-1} & & a_{n,j+1} & \cdots & a_{nn} \end{vmatrix}$，它们归并后所排成的 n-1 阶行列式$\begin{vmatrix} a_{21} & \cdots & a_{2,j-1} & a_{2,j+1} & \cdots & a_{1n} \\ a_{31} & \cdots & a_{3,j-1} & a_{3,j+1} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,j-1} & a_{n,j+1} & \cdots & a_{nn} \end{vmatrix}$，称为元素$a_{1j}$的余子式。

之前提到过，式$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$中，$a_{11}$，$a_{22}$，…，$a_{nn}$这些元素所在、所排成的对角线，称为行列式的主对角线。

1. 2 运用定义求特殊行列式的值&番外联想篇

1.番外联想篇

(1).有趣的符号扫雷图

由于我们在上一节的 4.中涉及到了代数余子式和余子式，其中的$A_{1j}=(-1)^{1+j}M_{1j}$在广义上写为$A_{ij}=(-1)^{i+j}M_{ij}$。也就是说，代数余子式和余子式的定义不局限于第一行的元

素a_{1j}，本质上对于任何一个第 i 行、第 j 列的元素a_{ij}来说，均有：M_{1j}为元素a_{ij}的余子式；A_{ij}为元素a_{ij}代数余子式。

由于行列式的展开式中用到的是各元素的代数余子式A_{ij}们，它们与对应的余子式M_{ij}们只相差一个符号$(-1)^{i+j}$，而M_{ij}又只在空间位置上定义(像数独一样，可以用“同类相斥”、“横竖消除”、“十字炸弹”的眼光来看待M_{ij}与a_{ij}的关系，保证M_{ij}的各元素的下标中：行标中无 i，列标中无 j)，属于非常好求的那类。那么问题就只剩下如何判断和求某元素a_{ij}所对应的符号$(-1)^{i+j}$了。

郭先生采用过的一个方法非常生动形象地介绍了：各位置上的元素a_{ij}所对应的代数余子式的符号——他说，$(-1)^{i+j}=\begin{vmatrix} + & - & + & \cdots \\ - & + & - & \cdots \\ + & - & + & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{vmatrix}$。这可被看为一种扫雷式的记忆方式：①.从左上角↖的“+”开始，每向着横向 or 竖向相邻的一格走一步，就改变奇偶性，进而改变正负号，用计算机语言来描述即为：先初始化 k=0，之后将A_{ij}的符号因子$=(-1)^{2+k}$；k=k±1 这两行，纳入一个循环结构中；②.当然，你还可以斜着走，此时 k=k±2，将新的两行语句放入无条件循环中后，会发现每向着对角线方向走入相邻一格，就改变正负号；③.当然，也不一定需要初始化为 k=0，这对应着你的初始格子不一定要选取左上角那个格。而可以在那个“+-+-八卦图”中随意选一点作为初始重生点，再走格子走到想要判断正负的格子处。

(2).逆序数的由来

①.Section1

假如现在让我们用定义法来求$\begin{vmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{vmatrix}$。根据如下规则：$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=a_{11}A_{11}+a_{12}A_{12}+\ldots+a_{1n}A_{1n}$，我们只用得上$A_{11}$、$A_{12}$、$A_{13}$、$A_{14}$，因而只需考虑这四个代数余子

式的正负号，所以使用的是$\begin{vmatrix} + & - & + & - \\ & & & \\ & & & \\ & & & \end{vmatrix}$(空出来的表示因不使用而不考虑的位置们)。

由于a_{11}、a_{12}、a_{14}都是 0，那么我们只需确定M_{13}前面的符号："+"；而现在仅剩的未知部分M_{13}，其性质与原行列式相同，也是一个完整且不掺杂的行列式：

$\begin{vmatrix} 0 & 1 & & 0 \\ 0 & 0 & & 1 \\ 1 & 0 & & 0 \end{vmatrix}$(空格表示没有元素存在，这是个三阶行列式)，那么我们继续往下降一阶。

一个有趣的现象发生了：根据$\begin{vmatrix} + & - & + & \cdots \\ - & + & - & \cdots \\ + & - & + & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{vmatrix}$，在原来的四阶行列式中，绿色符号处的符号们原本如此$\begin{vmatrix} + & - & + & - \\ - & + & - & + \\ + & - & + & - \\ - & + & - & + \end{vmatrix} = \begin{vmatrix} - & + & & + \\ + & - & & - \\ - & + & & + \end{vmatrix}$，但变成了现在降了一阶的三阶行列式

$\begin{vmatrix} + & - & & + \\ - & + & & - \\ + & - & & + \end{vmatrix} = \begin{vmatrix} + & - & + \\ - & + & - \\ + & - & + \end{vmatrix}$后，由于绿色符号所在的位置们上移了一格、左右不变(j=j、i=i-1)，所以其中的符号改变了；而红色符号们所在的位置既上移一格、又左移一格(j=j-1、i=i-1)，所以所有的红色符号均不改变。

可以证明，红绿灯区域遵从如下象限规则：$\begin{vmatrix} - & & + \\ + & & - \end{vmatrix}$，即一、三象限为绿灯区，符号改变属性为 positive，即在所划去的某行某列的右上角↗和左下角↙的区域内的符号因子，划去前的符号与划去后的符号异号；而二、四象限为红灯区，符号改变属性为 negative，即在所划去的某行某列的左上角↖和右下角↘的区域内的符号因子，在被划去的十字被划去后，符号因子的符号不变。该规则对任何阶数的行列式都适用。

这个规则(规律)能够保证任何阶数的行列式，在划去十字之前和划去十字之后，均以同样的扫雷分布图$\begin{vmatrix} + & - & + & \cdots \\ - & + & - & \cdots \\ + & - & + & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{vmatrix}$存在。虽然阶数不同，但它们是"相似的"，即所遵从的规律一致。

②.Section2

四个 1 也是数独般互斥的。

上一节所发现的规律简而言之：被划去的十字交叉点所在的元素，其次对角线方向↗↙的两片区域的元素们所对应的代数余子式的符号因子，会随着“划去一行一列”or“行列式降一阶”而改变符号，而其主对角线方向↖↘的两片区域的符号因子不会因“划去”而改变符号。——所以我们不必关心那些总处在接下来每一个将要被划去的十字交叉点格子的主对角线方向↖↘的两片区域的符号因子，它们的符号在最初的行列式所对应的最大的扫雷分布图$\begin{vmatrix} + & - & + & \cdots \\ - & + & - & \cdots \\ + & - & + & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{vmatrix} = \begin{vmatrix} + & - & + & - \\ - & + & - & + \\ + & - & + & - \\ - & + & - & + \end{vmatrix}$(←对于上一节中的例子来说，是这个)中，已经确定了。

也就是说，现在我们换个角度思考——不以某个将被划去的十字交叉点格子为主语，不以该格子为原点来建立笛卡尔坐标系，而将视线转移至那些被这个格子所建立的坐标系所衡量符号改变情况的四个区域中的需考虑的格子上来——由于我们所介绍的降阶规则是一阶一阶往下降的，且用的是每个M_{ij}中的最上面一行元素与其代数余子式的乘积之和。——所以整体来看是一行一行从上到下降阶计算的。——所以对于任意一个将要被考虑的格子来说，我们只需考虑其相对于其上面各行相继被划去的十字交叉元素们的相对位置，记录下其中其位于某十字交叉元素的次对角线方向的次数，该次数即为该被考虑的格子以其在最大的扫雷分布图中的符号为初始符号，改变符号的总次数。【根据乘法交换律，该种记录不需要考虑十字交叉元素们被划去的先后顺序】

同时，根据相对位置的相对性，若被考虑的格子相对于其上面的十字交叉元素们为次对角线方向，则其上面的十字交叉元素们相对于其也为次对角线方向。所以我们只需首尾连接其与其上面的十字交叉元素们，构成一条条斜线段(必定是斜的，不可能竖直

or 水平)，用它们的倾斜状况的计数来得到其符号改变的总次数=↗↙方向倾斜的斜线段个数。

利用“最大的扫雷分布图中的初始符号”以及“符号改变的总次数”，就可以得到被考虑的格子的最终符号$=(-1)^{\text{符号改变的总次数}}\times$初始符号。

③.Section3

对于 Section1 中的例子$\begin{vmatrix}0&0&1&0\\0&1&0&0\\0&0&0&1\\1&0&0&0\end{vmatrix}$来讲，需要被考虑的格子为 4 个红色的“1”。而第一行的“1”在头一行，其上面没有行了，所以其符号改变的总次数=**0**；第一行的“1”相对于第二行的“1”来说，是十字交叉元素。而第二行的“1”与第一行的“1”连线方向为↗↙次对角线方向，因此其符号改变的总次数=**1**；第一行的“1”和第二行的“1”都相继被划去，才到了第三行，因此对于第三行的“1”，它们均为它之前的十字交叉元素们，而它分别与它们的连线均为主对角线方向↖↘，因此其符号改变的总次数=**0**；同样的道理，在第四行的“1”与前三行相继被划去的十字交叉元素“1”相连的三段斜线段中，共计有 3 个↗↙方向倾斜的斜线段。因此其符号改变的总次数=**3**。

由于初始行列式$\begin{vmatrix}0&0&1&0\\0&1&0&0\\0&0&0&1\\1&0&0&0\end{vmatrix}$最终的值的正负号，由这 4 个被考虑的格子的最终符号之积所确定$=(-1)^{\sum_{i=1}^{4}\text{第 i 个被考虑格子的符号改变的总次数}}\times\prod_{i=1}^{4}$第 i 个被考虑格子的初始符号。

而我们已然得到了$\sum_{i=1}^{4}$第 i 个被考虑格子的符号改变的总次数=**0**+**1**+**0**+**3**=**4**，所以只需考虑剩下的这一项因子：$\prod_{i=1}^{4}$第 i 个被考虑格子的初始符号。

有意思的是，这项因子恒=+1。也就是说，在最大的扫雷分布图$\begin{vmatrix}+&-&+&-\\-&+&-&+\\+&-&+&-\\-&+&-&+\end{vmatrix}$中，任何一组个数与行列式阶数相同的、行序数互不相同且列序数互不相同的元素a_{ij}所构成的序列，其 4 个元素所对应的 4 个代数余子式A_{ij}的符号因子之积，一定为正。——证

明如下：利用定义$A_{ij}=(-1)^{i+j}M_{ij}$，知A_{ij}的符号因子$=(-1)^{i+j}$；由于各a_{ij}的行序数 i 互异，而a_{ij}的个数又=对应行列式的阶数=4，那么对于只能取 1,2,3,4 的 i，各a_{ij}的 i 必然取遍了 1,2,3,4。同样的道理，各a_{ij}的 j 也必然取遍了 1,2,3,4。相当于在 n 种互异的球中选 n 个互异的出来，则选出来的 n 个球中共计 n 种球，且全是互异的。

那么$\prod_{i=1}^{4}$第 i 个被考虑格子的初始符号$=(-1)^{\sum_{i=1}^{4}i+\sum_{j=1}^{4}j}=(-1)^{(1+4)\cdot 4}=1$。同样的道理，对于 n 阶行列式中的某一组有 n 个行序互异、列序互异的元素的序列而言，其$\prod_{i=1}^{n}$第 i 个被考虑格子的初始符号$=(-1)^{(1+n)\cdot n}=1$。可见该项因子恒为 1。所以对于一组 n 个行、列互异的元素，其所对应的 n 个被考虑的格子的最终符号之积

$=(-1)^{\sum_{i=1}^{n}\text{第 i 个被考虑格子的符号改变的总次数}}\times 1=(-1)^{\sum_{i=1}^{n}\text{第 i 个被考虑格子的逆序数}}=(-1)^{\text{总逆序数}}$。

其中，第 i 个被考虑格子的逆序数=n 个元素依行(列)序由小到大(由大到小)顺序排列，第 i 个元素的列(行)与前面共 i-1 个元素的列(行)做 i-1 次比较，结果为小于(大于)的次数=↗↙方向倾斜的斜线段个数=第 i 个被考虑格子的符号改变的总次数。

元素由行序小到行序大地排列(由大到小)→行序大的元素的列序却小于(大于)行序小的元素的列序：这便体现了“逆序数”中的“逆”。所以我们所常说的“逆序数”，指的便是这里的“总逆序数”，归根结底即为$\sum_{i=1}^{n}$第 i 个被考虑格子的符号改变的总次数。

④.Section4

我们回看整个过程，这一切都是由“用初始所给的定义法求$\begin{vmatrix}0&0&1&0\\0&1&0&0\\0&0&0&1\\1&0&0&0\end{vmatrix}$的值”这个问题引申而来。但这些所引申出来的东西，却不止步于仅仅解决了这一个小问题。——

我们现在将这个问题进一步拓宽到更广义的情形：如果我们现在要求$\begin{vmatrix}0&0&a_{13}&0\\0&a_{22}&0&0\\0&0&0&a_{34}\\a_{41}&0&0&0\end{vmatrix}$的值呢？其最终结果的正负号，是不是也是由$(-1)^{\text{总逆序数}}$决定？并且其正负情况和之前

相同？——再往前走一步，如果我们现在想知道$\begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix}$的结果——一个由 4×3×2×1(为什么是 4!呢，这是因为将原行列式的第一层展开式中的 4 个三阶展开，再将每个三阶的第一层展开式中的三个二阶展开…)个类似的项所组成的多项式，其中$a_{13}a_{22}a_{34}a_{41}$这一项的符号情况，其符号法则是不是也是：$(-1)^{总逆序数}$？并且，单独从$a_{13}a_{22}a_{34}a_{41}$这某一项来讲，$\begin{vmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{vmatrix} \sim \begin{vmatrix} 0 & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{vmatrix} \sim \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix}$，三者的结果中，这一项的符号是一样的，因为要知道，规则作用于位置，而他们位置相同；另一个原因：这是“线性”代数，$\begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix}$=4!个诸如$\begin{vmatrix} 0 & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{vmatrix}$的项之和。

当然，以上仅仅是个人对逆序数的得来的看法。真正用的时候，基本上都是将我们最终所得的规律总结来作为工具来使用，这是使用说明：对于一个给定的行列式，或者不需要给行列式，直接问其中的$a_{34}a_{41}a_{13}a_{22}$这项的符号是正 or 负。——首先，将行序小的元素排左边，行序大的元素排右边，得到$a_{13}a_{22}a_{34}a_{41}$，使得此时这组排好后的元素的行序列 1234 的逆序数$\tau(1234)$=0，即任何一个数字的右(左)边的数字都比它大(小)。将元素排列成这样后，再考虑其列序列 3241 的逆序数$\tau(3241)$=2+1+1+0。——我们从左至右取数(当然，对象的选取也可依其他爱好的规律，但得选完/不漏且不重复)，且从左向右比对：对于数字 3，3 比 2 大，3 比 1 大，记为 2 次；对于数字 2，2 比 1 大，加一次；对于数字 4,4 比 1 大；1 没列序可比。在设置好元素的排列依照对应的行序列升序排列后，对于列序列而言，共计 4 次“左边数字>右边数字”的反常事件(降序)发生，因此逆序数=4。

当然，操作过程因人而异，也有的人喜欢从右边的 1 开始取数来比较，看 1 比左边的 324 小的有多少对，再看 4，再看 2。也有的人从一开始喜欢左至右降序排列行(不

过一般行依左→右升序排列，对应行列式中行依上→下升序排列)，然后看同样左至右，有多少对升序排列的列序。当然，也有的人喜欢先排列，再比较行。——这些都是可取的，只要你的习惯最终帮你实现了目的、找出了总的逆序数。——当然，当心你需要不多不少、老老实实、明暗相间地比较$C_n^2=C_4^2=6$次。

⑤.Section5

另一种也简单可行的求等效总逆序数的方法，我叫它“swap 大法”：

之前第一步是规定从小到大为标准次序，把元素从左到右 or 从右到左依照行的标准顺序排列，现在我们没有这一步了。之前的第二步是以相同方向察看“逆着标准顺序”的列序有多少对——而现在是将列序列中任意两个列序进行位置调换，同时记录下调换的次数，直到列序列变成与行序列相同的序列(不一定是标准顺序)，停止调换。此时，等效总逆序数 = 总共调换(swap)的次数， 且$(-1)^{\text{总逆序数}}=(-1)^{\text{等效总逆序数}}=(-1)^{\text{swap 的总次数}}$。

举个例子，并将其中的原因先如下简单介绍一下：对于$a_{34}a_{41}a_{13}a_{22}$来说，可发现其行序列 3412 不是标准序列 1234 或 4321，我们不需要理睬它。将其列序列 4132 提取出来，想办法通过两两互换(swap)的方式，让其变为 3412。

一个比较简单的做法是：先将列序列 4132 中，$3\boxed{\leftrightarrow}4$ 互换，得到 3142(对应的$a_{34}a_{41}a_{13}a_{22}$变为$a_{33}a_{41}a_{14}a_{22}$；对应的$\begin{vmatrix}0&0&a_{13}&0\\0&a_{22}&0&0\\0&0&0&a_{34}\\a_{41}&0&0&0\end{vmatrix}$变为$\begin{vmatrix}0&0&0&a_{14}\\0&a_{22}&0&0\\0&0&a_{33}&0\\a_{41}&0&0&0\end{vmatrix}$)；再将 3142 中，$1\boxed{\leftrightarrow}4$ 互换，得到 3412(对应的$a_{33}a_{41}a_{14}a_{22}$变为$a_{33}a_{44}a_{11}a_{22}$；对应的$\begin{vmatrix}0&0&0&a_{14}\\0&a_{22}&0&0\\0&0&a_{33}&0\\a_{41}&0&0&0\end{vmatrix}$

变为$\begin{vmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{vmatrix}$)。

可以发现，若利用乘法交换律将列序和行序均为 3412 的$a_{33}a_{44}a_{11}a_{22}$这些元素按照行序 or 列序标准排列(1234)：$a_{11}a_{22}a_{33}a_{44}$，就可清楚地看到，$a_{11}a_{22}a_{33}a_{44}$作为$\begin{vmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{vmatrix}$的最终结果，其总逆序数=0，其符号=$(-1)^0$恒为正。

现在我们将整个过程逆向地复现一遍，从$a_{33}a_{44}a_{11}a_{22}$回到$a_{34}a_{41}a_{13}a_{22}$，也就是说从$\begin{vmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & a_{33} & 0 \\ 0 & 0 & 0 & a_{44} \end{vmatrix}$回到$\begin{vmatrix} 0 & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{vmatrix}$，若照搬之前过程的逆过程，先进行列的 1$\boxed{\leftrightarrow}$4 互换，再列的 3$\boxed{\leftrightarrow}$4 互换，则需要经过 2 步 swap 才能回到初始；而若尝试用其他你还可以想到的交换顺序和方法，则会经过 2+2k 次 swap。也就是说，至少经过 2 次、总会经过偶数次地，swap 后，你才能从 3412 回到 4132、或是从 4132 去往 3412。

事实上，单独每一次 swap，都会改变$(-1)^{总逆序数}$的指数，即总逆序数的奇偶性(总逆序数=总逆序数±奇数)，进而改变诸如$\begin{vmatrix} 0 & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{vmatrix} \rightarrow (-1)^{总逆序数} \cdot a_{34}a_{41}a_{13}a_{22}$中符号因子$(-1)^{总逆序数}$的正负。而考虑列序列从 3412 回到 4132 的过程，我们是从$(-1)^0$开始的，即先初始化总逆序数=0，每 swap 一次，总逆序数=总逆序数±奇数——等效于先初始化等效总逆序数=0，每 swap 一次，等效总逆序数=等效总逆序数+1。这两个分支在经历了同样的 swap 次数后，所得的最终结果是相等的：$(-1)^{总逆序数}=(-1)^{等效总逆序数}=(-1)^{swap\text{ 的总次数}}$。所以我们可以直接用swap 的总次数作为符号因子的最终指数。

由于从 4132 去往 3412 和从 3412 回到 4132 的 swap 次数是相等或者相差偶数次

的，因此我们用从 3412 回到 4132 的过程来解释从 4132 去往 3412 的结果，是可行的。并且这同时也提醒我们，在规则上，我们也可以：将行序列中任意两个行序进行位置调换，同时记录下调换的次数，直到行序列变成与列序列相同的序列(不一定是标准顺序)，停止调换。这种意义下的 swap 次数作为符号因子的指数也是可行的。

所以该种“Swap 大法”的操作步骤已经解释清楚了，连同其之所以可行的大部分原因。其运作机理中，唯一尚未解释的是，为什么每一次 swap，都会不多不少地恰好 100%改变总逆序数的奇偶性。这将用到之后所介绍的，行列式的性质及其推论之一：“如果行列式两行互换，行列式改变符号。”

Swap，微观上，行列式的完全展开式的每一项，都可视为子行列式，都会因 swap 某两行 or 两列，而改变符号(而子行列式因 swap 而变号的现象，就是每一次 swap 均会改变总逆序数的等效说法)；因此宏观上作为 n!个行列式项之和的 D 会变成-D。到时候我们也会用那里的理论直接宏观操作得到这个“交换两行 or 两列对应 D→-D 事件的发生”的结论，并用该结论证明每个子行列式也遵从该结论。

2.运用定义求特殊行列式的值

(1).一些常用的简单行列式的值

①.$D_n=\begin{vmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{vmatrix}$，这样的 n 阶行列式，称为 n 阶对角行列式。(一般来说，这种情况下，行列式中有些格子没有写元素，表示该格子中的元素值为 0；我在编写时，甚至把格子都给去掉了，这种操作叫“隐藏占位符”)

根据行列式的原始定义，$D_n=\begin{vmatrix} a_1 & & & \\ & a_2 & & \\ & & \ddots & \\ & & & a_n \end{vmatrix}=a_{11}\begin{vmatrix} a_2 & & & \\ & a_3 & & \\ & & \ddots & \\ & & & a_n \end{vmatrix}=\ldots=a_1a_2\cdots a_n$。因此，对角行列式的值=主对角线元素之积，即$D_n=a_1a_2\cdots a_n$。【这里的$a_i$相当于$a_{ii}$，为了简单起见，就用的是$a_i$】

②. $D_n=\begin{vmatrix} a_{11} & & & \\ a_{21} & a_{22} & & \\ \vdots & \vdots & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$称为(n 阶)下三角形行列式。

同样根据定义，$D_n=\begin{vmatrix} a_{11} & & & \\ a_{21} & a_{22} & & \\ \vdots & \vdots & \ddots & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=a_{11}\begin{vmatrix} a_{22} & & & \\ a_{32} & a_{33} & & \\ \vdots & \vdots & \ddots & \\ a_{n2} & a_{n3} & \cdots & a_{nn} \end{vmatrix}=\ldots=a_{11}a_{22}\cdots a_{nn}$。因此，下三角行列式的值也=主对角线元素之积，即$D_n=a_{11}a_{22}\cdots a_{nn}$。

③.由于我们已经用了“对角行列式”这个来默认代表“主对角行列式”，所以要叫它的话，接下来的这个只能被叫做“次对角行列式”了：

根据行列式的定义，$D_n=\begin{vmatrix} & & & b_n \\ & & b_{n-1} & \\ & \iddots & & \\ b_1 & & & \end{vmatrix}=(-1)^{n+1}\cdot b_n\cdot D_{n-1}=(-1)^{n-1}\cdot b_n\cdot D_{n-1}=(-1)^{n-1}\cdot b_n\cdot(-1)^{n-2}\cdot b_{n-1}\cdot D_{n-2}=(-1)^{n-1}\cdot b_n\cdot(-1)^{n-2}\cdot b_{n-1}\cdots(-1)^2\cdot b_3\cdot D_2=(-1)^{n-1}\cdot b_n\cdot(-1)^{n-2}\cdot b_{n-1}\cdots(-1)^2\cdot b_3\cdot(-1)\cdot b_2b_1=(-1)^{\frac{n(n-1)}{2}}\cdot b_n\cdot b_{n-1}\cdots b_1$。

现在我们来关注其系数$(-1)^{\frac{n(n-1)}{2}}$：令n=4k+a，a=0,1,2,3。当a=0或1时，$(-1)^{\frac{n(n-1)}{2}}=(-1)^{2k(4k-1)}$或$(-1)^{2k(4k+1)}=[(-1)^{4k\pm1}]^{2k}=+1$；当 a=2 或 3 时，$(-1)^{\frac{n(n-1)}{2}}=(-1)^{(2k+1)(4k+1)}$或$(-1)^{(2k+1)(4k+3)}=[(-1)^{4k+1\text{or}+3}]^{2k+1}$=-1。奇数次幂保持原号，偶数次幂为正号。

1.3 行列式的性质

在上一节我们提到过，任何一个 n 阶行列式的最终展开式，这个多项式有 n!项，且

每一项都可看作一个像这样$\begin{vmatrix} 0 & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & 0 \\ 0 & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{vmatrix}$的同阶的子行列式。每个子行列式值为 n 个元素的乘积，需要做 n-1 次乘法运算；而又有 n!那么多个子行列式相加，所以利用最初的定义法来求解一个 n 阶行列式，总共需要做 n!-1 次加法运算，以及(n-1)·n!次乘法运算，计算量是非常可观的(可以用斯特林公式做阶乘的近似计算)。

然而上一节中，我们计算对角行列式、三角行列式，计算量并不大；所以我们希望推导出行列式的一些性质，将一般行列式尽量化简为我们可以人工心算的这些熟悉的行列式，用以快速和简便计算复杂行列式的值。

1.转置 transpose

$D^T=\begin{vmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{nn} \end{vmatrix}$可以看为是由 $D=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$的每个元素们，以 D 的主对角线为轴翻转 180°所得。也可从数学上解释为，每个本身处于(i,j)的元素a_{ij}，都到新位置(j,i)上去了；而原先处于(j,i)这个格子框框里的元素a_{ji}，跑到了(i,j)位置处的格子中，虽然这后一句话已经被分号前的那句解释过而不需要书写了。

(1).性质 1　行列式与其转置行列式相等，即 $D=D^T$

①.证 part1

利用数学归纳法：当 n=2、3 时可计算出$D^T=D$ 成立，则假设$D^T=D$ 对阶数<n 的行列式都成立，看能不能推出 n 阶也成立：

根据定义，$D=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=a_{11}A_{11}+a_{12}A_{12}+\ldots+a_{1n}A_{1n}$，之前我们只能像这样

将 D **按第一行展开**；现我们尝试着将其(形式上地)**按第一列展开**，$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=$ $a_{11}A_{11}+a_{21}A_{21}+\ldots+a_{n1}A_{n1}$，先不管这里的等号是否成立，即 D 是否能这样展开；这一步只是为了介绍：**仍定义**其中的$A_{j1}=(-1)^{j+1}M_{j1}$，为 **D 中**的元素a_{j1}的代数余子式。而M_{j1}也为 **D 中**的元素a_{j1}的余子式=将 **D** 划掉第 j 行和第 1 列的所有元素后，剩余元素间的相对位置不变，归并后所排成的 n-1 阶行列式。

现在我们将D^T以原始的定义，**按第一行展开**，$D^T=\begin{vmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{nn} \end{vmatrix}=a_{11}A_{11}^T+a_{21}A_{21}^T$ $+\ldots+a_{n1}A_{n1}^T$。其中的$A_{j1}^T=$**D^T中**位于第 1 行和第 j 列的元素a_{j1}的代数余子式=**D 中**位于第 j 行和第 1 列的元素a_{j1}的代数余子式**的转置**。

而由于归纳法假设了$D^T=D$ 对阶数<n 的行列式都成立，因此在D^T的**第一行展开式**中，n 个 n-1 阶行列式均满足：$A_{11}^T=A_{11}$、$A_{21}^T=A_{21}\ldots$；因此$D^T=a_{11}A_{11}^T+a_{21}A_{21}^T+\ldots+a_{n1}A_{n1}^T=a_{11}A_{11}+a_{21}A_{21}+\ldots+a_{n1}A_{n1}$=D **按第一列展开**的结果。也就是说，由于$D_{n-1}^T=D_{n-1}$存在于数学归纳法的假设之中，可当成已知来用，因此根据这归纳假设的前面部分，如下结论在归纳法的假设成立时就已经成立："D^T=D **按第一列展开**的结果"(但暂时还无法说明"D=D **按第一列展开**的结果"，即"D 能**按第一列展开**")——而我们还需证明：D **按第一列展开**的结果=D **按第一行展开**的结果=D，才能证明$D^T=D$；或者，若我们通过其他途径独立证明了$D^T=D$，那么我们便同时证明了 D 可以像之前一样**按第一列展开**，且展开结果与其**按第一行展开**所得的一致。

事实上，我们暂且没法通过上文中的"其他途径"来独立证明$D^T=D$。我们手中只拥有**按行展开的规则**，以及在归纳法所假设的$D_{n-1}^T=D_{n-1}$成立时，将成立的 n 阶及以下的 D 均满足："D^T=D **按第一列展开**的结果"。我们将用它俩来得到"D **按第一列展开**的结果=D **按第一行展开**的结果"，进而得到$D^T=D$。

②.证 part2

根据上个部分的引入，$D^T=D$ **按第一列展开**的结果 $=a_{11}A_{11}+a_{21}A_{21}+\ldots+a_{n1}A_{n1}=\sum_{i=1}^{n}a_{i1}A_{i1}$，现将 n 个$A_{i1}$中，除了$A_{11}$之外的 n-1 个 n-1 阶行列式都按照定义展开(**按第一行展开**)，并将其中含a_{12}、$a_{13}\ldots a_{1n}$的项，分门别类地合并在一起，即有：$\sum_{i=1}^{n}a_{i1}A_{i1}=a_{11}A_{11}+\sum_{i=2}^{n}a_{i1}A_{i1}=a_{11}A_{11}+\sum_{j=2}^{n}a_{21}a_{1j}A_{21_{1j}}+\sum_{j=2}^{n}a_{31}a_{1j}A_{31_{1j}}+\ldots+\sum_{j=2}^{n}a_{n1}a_{1j}A_{n1_{1j}}=a_{11}A_{11}+\sum_{i=2}^{n}\sum_{j=2}^{n}a_{i1}a_{1j}A_{i1_{1j}}=a_{11}A_{11}+\sum_{i=2}^{n}a_{i1}a_{12}A_{i1_{12}}+\sum_{i=2}^{n}a_{i1}a_{13}A_{i1_{13}}+\ldots+\sum_{i=2}^{n}a_{i1}a_{1n}A_{i1_{1n}}=a_{11}A_{11}+a_{12}\sum_{i=2}^{n}a_{i1}A_{i1_{12}}+a_{13}\sum_{i=2}^{n}a_{i1}A_{i1_{13}}+\ldots+a_{1n}\sum_{i=2}^{n}a_{i1}A_{i1_{1n}}=a_{11}A_{11}+\sum_{j=2}^{n}a_{1j}A_{1j}=\sum_{j=1}^{n}a_{1j}A_{1j}$。——其中，正如$A_{21}=\sum_{j=2}^{n}a_{1j}A_{21_{1j}}$地，对称地有$A_{12}=\sum_{i=2}^{n}a_{i1}A_{i1_{12}}$；$A_{13}$也仿照$A_{31}$的展开，对$A_{1j}$的操作诸如此类。

$\sum_{i=2}^{n}a_{i1}A_{i1_{12}}$能凑成$A_{12}$的深刻原因如下：细心观察$A_{1j}$的展开式，可发现他们与$A_{i1}$**按第一行展开的**展开式不同，全是**按第一列展开的**，而其展开式中共 n-2 个诸如$A_{21_{1j}}$的子行列式全是 n-2 阶的，而由于数学归纳法的荫庇，$D^T=D$ 对阶数<n 的行列式都成立，因此“$D^T=D$ **按第一列展开**的结果”也适用于 D 的阶数为 n-1 的情况。——于是$\sum_{i=2}^{n}a_{i1}A_{i1_{12}}=D_{n-1}$**按第一列展开**的结果$=D_{n-1}^T=A_{12}^T$；而又因数学归纳法的假定，$D_{n-1}^T=D_{n-1}$，所以$A_{12}^T=A_{12}$。所以$\sum_{i=2}^{n}a_{i1}A_{i1_{12}}=A_{12}^T=A_{12}$。

综上，$D^T=\sum_{i=1}^{n}a_{i1}A_{i1}=\sum_{j=1}^{n}a_{1j}A_{1j}=D$。至此我们即完成了利用$D_1^T=D_1$、$D_{n-1}^T=D_{n-1}$得到$D_n^T=D_n$的过程。也就完成了对$D^T=D$ 这个结论的自洽性的证明。并且，利用性质中的$D_n^T=D_n$和$D_{n-1}^T=D_{n-1}$，可得 $D=D^T=D$ **按第一列展开**的结果，即之前所预言的那样：D 可以**按第一列展开**，且不用担心地，结果将与其**按第一行展开**所得的相同。——更广地，不仅之前的定义式(按行逐阶降阶法则)，之后的几条性质也不仅针对行，列同样具有行的性质和展开方式。——对 D 的行的操作，等效于对D^T的相应的列的操作(像 $D\to D^T$的映射法则一样，“操作”也得遵循这样的轴对称法则)，而 $D=D^T$，所以对 D 的行的操作

=对 D 的列的操作。因此，之后关于行列式的性质可以仅对行叙述。

例 1.1 D= $\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22} & \cdots & a_{2n} \\ & & \ddots & \vdots \\ & & & a_{nn} \end{vmatrix}$ 这样一个主对角线之下的元素都是 0 的行列式，叫做上三角形行列式。

你既可以通过“先将D^T这个下三角形行列式，每次都**按第一行展开**(or 直接沿用关于下三角形行列式的结论)，再用 D=D^T”；又可以通过“对 D 一直**按第一列展开**”来直接得到：D=$a_1 a_2 \cdots a_n$。

(2).性质 2 对行列式中的任一行按下式展开，其值相等，即等于行列式的值

$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} = a_{i1}A_{i1} + a_{i2}A_{i2} + \ldots + a_{in}A_{in}$，其中$A_{ij} = (-1)^{i+j}M_{ij}$，称为元素$a_{ij}$的代数余子式。同样，$M_{ij} = \begin{vmatrix} a_{11} & \cdots & a_{1,j-1} & a_{1,j+1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{i-1,1} & \cdots & a_{i-1,j-1} & a_{i-1,j+1} & \cdots & a_{i-1,n} \\ a_{i+1,1} & \cdots & a_{i+1,j-1} & a_{i+1,j+1} & \cdots & a_{i+1,n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,j-1} & a_{n,j+1} & \cdots & a_{nn} \end{vmatrix}$，称为元素$a_{ij}$的余子式。

①.证

同样利用数学归纳法：当 n=2 时可计算出结论成立，假设结论对 n-1 阶行列式成立，下面考虑 n 阶的情况：

根据原始定义将 D **按第一行展开**：D=$\sum_{j=1}^{n} a_{1j}A_{1j}$，再根据归纳法的假设，将其中 n 个 n-1 阶行列式都按“**它们的第 i-1 行**”展开(这些行对于大 D 来说，是第 i 行)，并将其中含有a_{i1}、a_{i2}...a_{in}等不同 J 的a_{iJ}的项分别单独归为一类，合并在一起：D=$\sum_{j=1}^{n} a_{1j}A_{1j}$=

$\sum_{j=1(j\neq 1)}^{n} a_{11}a_{ij}A_{11_{ij}} + \sum_{j=1(j\neq 2)}^{n} a_{12}a_{ij}A_{12_{ij}} + \ldots + \sum_{j=1(j\neq n)}^{n} a_{1n}a_{ij}A_{1n_{ij}} = \sum_{J=1}^{n}\sum_{j=1(j\neq J)}^{n} a_{1J}a_{ij}A_{1J_{ij}} =$

$\sum_{j=1}^{n}\sum_{J=1(J\neq j)}^{n} a_{ij}a_{1J}A_{ij_{1J}}=\sum_{J=1(J\neq 1)}^{n} a_{i1}a_{1J}A_{i1_{1J}}+\sum_{J=1(J\neq 2)}^{n} a_{i2}a_{1J}A_{i2_{1J}}+\dots+\sum_{J=1(J\neq n)}^{n} a_{in}a_{1J}A_{in_{1J}}=$ $\sum_{j=1}^{n} a_{ij}A_{ij}$。其中$A_{11}=\sum_{j=1(j\neq 1)}^{n} a_{ij}A_{11_{ij}}$**是按**$A_{11}$**的第 i-1 行展开**的，符合我们的假设：**n-1 阶行列式可按其任意一行展开**；而$A_{i1}=\sum_{J=1(J\neq 1)}^{n} a_{1J}A_{i1_{1J}}$**是按**$A_{i1}$**的第 1 行展开**的，是自然而然成立的，所以$\sum_{J=1(J\neq 1)}^{n} a_{1J}A_{i1_{1J}}=A_{i1}$。

综上 D=$\sum_{j=1}^{n} a_{1j}A_{1j}=\sum_{j=1}^{n} a_{ij}A_{ij}$，即我们证明了，$D_n$**能按其第 i 行展开**；即结论“行列式可按任意一行展开”对 n 阶行列式同样成立。——有趣的是，我们在这两个性质中的证明，均是对“行列式的阶数”采用的数学归纳法，而不是对其他采用。比如，证明**性质 2**，能不能对“行列式的行数”采用数学归纳法呢？第 1 行、第 2 行、第 n-1 行、第 n 行？我觉得不行。在之后的其他行列式的性质的证明中，或者更远地，在之后的一些不容易证明的证明中，很多都是对“行列式的阶数”，进行数学归纳法。

例 1.2 根据性质 2，既然 D 能按任意行展开，再利用性质 1，则 D 还能按任意列展开；又根据我们的经验(比如三角形行列式)，更希望对拥有更多 0 元素的某行 or 某列进行展开。——那么问题来了，利用现有的知识，怎样最简单地计算它：$\begin{vmatrix}0&2&0&3\\1&-1&2&0\\0&2&0&0\\3&0&5&9\end{vmatrix}$？

我觉得你所想的和我一样：$\begin{vmatrix}0&2&0&3\\1&-1&2&0\\0&2&0&0\\3&0&5&9\end{vmatrix}\to -2\cdot\begin{vmatrix}0&&0&3\\1&&2&0\\&&&\\3&&5&9\end{vmatrix}\to -2\cdot\begin{vmatrix}0&0&3\\1&2&0\\3&5&9\end{vmatrix}\to -6\cdot\begin{vmatrix}&&\\1&2&\\3&5&\end{vmatrix}\to -6\cdot\begin{vmatrix}1&2\\3&5\end{vmatrix}\to -6\cdot(5-6)=6$。

(3).性质 3 行列式某行有公因子可以按行提取公因子，即：

$$\text{第 i 行}\to\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\k\cdot a_{i1}&k\cdot a_{i2}&\cdots&k\cdot a_{in}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}=k\cdot\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\a_{i1}&a_{i2}&\cdots&a_{in}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}。$$

①.证

利用性质 2，将等号左边的行列式**按第 i 行展开**，往左提出展开所得 n 项多项式的每项的 k，再将右边的多项式“合拢”，写成行列式的样子。

推论 行列式中某一行元素全为 0 时，行列式为 0。

(4).性质 4 行列式的拆行相加性，即：

$$\text{第 i 行}\rightarrow\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ b_1+c_1 & b_2+c_2 & \cdots & b_n+c_n \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ b_1 & b_2 & \cdots & b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}+\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & \cdots & c_n \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$$

,其中等号左边的b_i+c_i们、等号右边的b_i们、c_i们，所在的行，均为同一行。

①.证

同样，将等号两边的行列式都**按第 i 行展开**，重新归类即可。

(5).性质 5 行列式两行对应元素全相等，则该行列式为零，即：

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=0$$

，其中$a_{kj}=a_{lj}$，$k\neq l$。

①.证

同样利用数学归纳法：当 n=2 时可计算出结论成立，假设结论对 n-1 阶行列式成立，下面考虑 n 阶的情况：

只要 $n\geq 3$，则D_n必然有除了两个“同列元素对应相同”的行之外的行存在，则只

需要对D_n按第 i 行(i≠k,l)展开即可：其展开式中的 n 个 n-1 阶(余)子式M_{ij}，必然均包含第 k 行和第 l 行，因此根据归纳法的假设，各个M_{ij}、A_{ij}均=0，所以 D=其和=0。

推论 1 行列式两行对应元素成比例，行列式为 0。利用性质 3 和性质 5 即得。

推论 2 设行列式 $D=\left|a_{ij}\right|_n$，则$\sum_{j=1}^{n} a_{kj}A_{lj}=a_{k1}A_{l1}+a_{k2}A_{l2}+\ldots+a_{kn}A_{ln}=D\cdot\delta_{kl}$，其中$\delta_{kl}=1$, k=l; =0, k≠l, 称为 Kronecker(克罗内克)符号。【$\left|a_{ij}\right|_n$←这只是个该死的符号，用以代表行列式的各种展开规则所等于的行列式值，同时显示 D 是个行列式、D 的阶数、D 的基本元素符号用 a 表示等，将要被提及 or 引用的信息】

当 k=l 时，该推论即在描述性质 2。当 k≠l 时，在$\sum_{j=1}^{n} a_{kj}A_{lj}=a_{k1}A_{l1}+a_{k2}A_{l2}+\ldots+a_{kn}A_{ln}$中，(不同 j 所对应的)各$a_{kj}$的值，与各$a_{lj}$的值无关；而各$A_{lj}$的值，在定义上本身就与“处在各个(l,j)位置上的格子中的”各a_{lj}的值无关，所以在整体上，$\sum_{j=1}^{n} a_{kj}A_{lj}$的值与各$a_{lj}$的值无关。因而我们可以将 k≠l 时的$\sum_{j=1}^{n} a_{kj}A_{lj}$写作：$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$，其中无论第 l 行的各元素$a_{lj}$们取何值，都不影响$\sum_{j=1}^{n} a_{kj}A_{lj}$的最终结果=0，因为我们可以将各个$a_{lj}$取值为对应 j 下的$a_{kj}$，此时就回到了性质 5：$\sum_{j=1}^{n} a_{kj}A_{lj}=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}=0$。

该推论说明，行列式某一行的元素与另一行的元素的代数余子式对应乘积之和为 0。以上对行的结果可自然地对列描述：设 $D=\left|a_{ij}\right|_n$，则$\sum_{i=1}^{n} a_{ik}A_{il}=a_{1k}A_{1l}+a_{2k}A_{2l}+\ldots+a_{nk}A_{nl}=D\cdot\delta_{kl}$。

例 1.3 一些由性质 5 的推论 2 引申出来的小技巧：“位置型定义”的精髓所在：

给你一个行列式$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}$，请问$A_{22}+A_{23}=$? 又如问$M_{31}+M_{32}+M_{33}=$? ——若按照定义，算了 2 个或 3 个二阶行列式之后，再将其相加减，岂不是很麻烦？——不如这样

脑补一些没有的过程出来：$A_{22}+A_{23}=0\cdot A_{21}+1\cdot A_{22}+1\cdot A_{23}=a_{21}\cdot A_{21}+a_{22}\cdot A_{22}+a_{23}\cdot A_{23}$，其中$a_{21}=0$、$a_{22}=1$、$a_{23}=1$，因此$A_{22}+A_{23}$就被改造成了一个新的行列式的值$=\begin{vmatrix}1&2&3\\0&1&1\\7&8&9\end{vmatrix}$。

同样的道理：$M_{31}+M_{32}+M_{33}=A_{31}-A_{32}+A_{33}=1\cdot A_{21}+(-1)\cdot A_{22}+1\cdot A_{23}=\begin{vmatrix}1&2&3\\4&5&6\\1&-1&1\end{vmatrix}$。

我们将其凑成了个高阶行列式的值，而这样做之后就剩如何简便计算它了，下面的性质 6 就旨在服务于此。

(6).性质 6 行列式某行元素的若干倍加到另一行对应元素所得行列式值不变，即：

$$\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\a_{k1}&a_{k2}&\cdots&a_{kn}\\\vdots&\vdots&\ddots&\vdots\\a_{l1}&a_{l2}&\cdots&a_{ln}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}=\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\a_{k1}&a_{k2}&\cdots&a_{kn}\\\vdots&\vdots&\ddots&\vdots\\b\cdot a_{k1}+a_{l1}&b\cdot a_{k2}+a_{l2}&\cdots&b\cdot a_{kn}+a_{ln}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}\ (k\neq l)$$

①.证

利用性质 4 和性质 5 的推论 1 即得。

(7).性质 7 如果行列式两行互换，行列式只改变符号，即：

$$\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\a_{k1}&a_{k2}&\cdots&a_{kn}\\\vdots&\vdots&\ddots&\vdots\\a_{l1}&a_{l2}&\cdots&a_{ln}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}=-\begin{vmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\\vdots&\vdots&\ddots&\vdots\\a_{l1}&a_{l2}&\cdots&a_{ln}\\\vdots&\vdots&\ddots&\vdots\\a_{k1}&a_{k2}&\cdots&a_{kn}\\\vdots&\vdots&\ddots&\vdots\\a_{n1}&a_{n2}&\cdots&a_{nn}\end{vmatrix}\ (k\neq l)$$

①.证

接下来每一个(除最后一个)等号成立的原因，都将是性质 6: $\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$→(-1)*第 k 行，之积加到第 l 行(row)上去；计算机语言为$r_l=r_l+(-1)\cdot r_k$；数学语言简化为$r_l+(-1)\cdot r_k=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{k1}+a_{l1} & -a_{k2}+a_{l2} & \cdots & -a_{kn}+a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$→$r_k+r_l$(表示$r_k=r_k+r_l$，主语是$r_k$)$=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{k1}+a_{l1} & -a_{k2}+a_{l2} & \cdots & -a_{kn}+a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$→$r_l-r_k=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{k1} & -a_{k2} & \cdots & -a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$→性质 3$=-\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{l1} & a_{l2} & \cdots & a_{ln} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$。

同样，为了具体表述行列式的计算过程，可同样采用类似上述符号的，如：

1.交换第 k,l 两行(列)的位置：$r_k\leftrightarrow r_l$ $(c_k\leftrightarrow c_l)$；计算机语言版：x=r_k;r_k=r_l;r_l=x。其中 c：column。——性质 7

2.第 k 行(列)提出公因子 b：$r_k\div b$ $(c_k\div b)$；计算机语言版：$r_k=r_k\div b$。——性质 3

3.将第 k 行(列)的 b 倍加到第 l 行(列)：$r_l+b\cdot r_k$ $(c_l+b\cdot c_k)$；计算机语言版：$r_l=r_l+b\cdot r_k$。——性质 6

这些规则建立在行列式的性质之上。以上 7 个基本性质，它们对于计算、分析行列式是很有帮助的。在具体计算行列式问题时，需要根据行列式的不同特点而采取不同的方法，得灵活运用行列式的性质。

(8).性质 8 拉普拉斯(Laplace)展开定理

在 n 阶行列式 D 中选定 k 个行(1≤k<n)，则行列式 D=由这 k 个行产生的所有 k 阶子式与它们相应的代数余子式对应乘积之和。

k 阶子式：在行列式 D 中取定 k 个行，第i_1,第i_2,…第i_k行($1\le i_1<i_2<\ldots<i_k\le n$)，再取定 k 个列：第$j_1$,第$j_2$,…第$j_k$列($1\le j_1<j_2<\ldots<j_k\le n$)，位于这些取定的行、列交叉点上的元素共有$k^2$个，保持它们彼此之间相互位置关系不变，所构成的一个 k 阶行列式，记为 N，称为 D 的一个 k 阶子式。

N 的余子式：在行列式 D 中去掉上面取定的 k 个行及 k 个列，余下的 n-k 个行与 n-k 个列上的元素保持彼此相互位置关系不变而构成的 n-k 阶行列式 M，叫做 N 的余子式。(N_i在地位上类似于之前的a_i)

N 的代数余子式：$A=(-1)^{(i_1+i_2+\cdots+i_k)+(j_1+j_2+\cdots+j_k)}\cdot M$，称为 N 的代数余子式。(A、M 这俩字符和含义与之前的无异，算是继承；不过由于现在a_i拓展成了N_i，从代数字母晋升为了行列式，因此现在 N 与 M 平起平坐了)

在 n 阶行列式中取定 k 个行后，由这 k 个行构成的所有 k 阶子式应该有C_n^k个，那么 n 阶行列式按 Laplace 展开定理展开，就应该有C_n^k项。看上去总要比 k=1 或 k=n-1 所对应的性质 2 中按某一行展开的计算量更大，但事实上这个晚来的定理在处理有些问题上将不会那么多余，相反它会大显身手。

Let’s have a closer look at C_n^k：之前我们说过，计算一个 n 阶行列式需要做(n-1)·n!次乘法运算，这是因为同一个 n 阶行列式，按照不同的展开方法展开到最后，总会得到相同的 n!那么多个项相加。我们来证明一下 Laplace 展开定理也符合这种规范——对一个 n 阶行列式选定 k 行，并按 Laplace 展开定理展开，展开式中共有C_n^k个项，而每个项均由一个 k 阶子式和一个 n-k 阶余子式的乘积构成。对于一个 k 阶行列式，其最终展

开式为 k!个项之和，它与另(n-k)!个项之和，的乘积，即为C_n^k个项中，每个项的子项总数。因此一个 n 阶行列式按照 Laplace 展开定理，也会最终产生$C_n^k \times k! \times (n-k)! = n!$个项，这与我们之前所得到和所运用的结论一致，一方面说明了 Laplace 展开定理在这方面是与旧理论相容的。

另一方面，这也是为什么老家伙们能突如其来地来了灵感，写下了这个定理。——他们很可能是通过“拆”：将既得结论 n!给拆成$C_n^k \times k! \times (n-k)!$，n!都已经是排列组合的东西了，为什么不想到$C_n^k = \frac{n!}{k! \times (n-k)!}$呢，再说它的分子不恰好就是需拼凑的 goal 么；再充分一点的理由，对于性质 2，脑子转的快的人或许直接通过 n×(n-1)!得到了 n!，就没再想下去了；但那些老头脑子故意转的稍微慢点，理了理思路，加了个 1，整个思路就成为另一条 countryroad 了：$n \times (n-1)! \times 1! = C_n^1 \times (n-1)! \times 1! = n!$。于是他们便进一步在 1!上做文章，将其拓展为了 k!，这便是 Laplace 展开定理“开发历程”的可能的版本之一。——并且，行列式、及其性质 2 等 7 条性质已经够神奇了，但有些人还觉得上帝有所隐瞒，特别是性质 2，即使它已经能做到任意行、任意列，但总是“一阶一阶地往下展开”，没法自由自在地“从中间某个地方”向两边展开，或者“倒着展开”。于是他们就寻思着得到了这个更灵魂的“行列式世界法则”。

另外，如果 Laplace 展开定理成立，那么如下结论必须成立：$(-1)^{i+j} = (-1)^{\left(\frac{n(n+1)}{2}-i\right)+\left(\frac{n(n+1)}{2}-j\right)} = (-1)^{n(n+1)-i-j}$。虽然没有 Laplace 展开定理，它也因$(-1)^{2i+2j} = (-1)^{n(n+1)} = 1$而成立。——你可以通过选定某 1 个行做 Laplace 展开，再选定其余 n-1 个行做 Laplace 展开，两个展开式的对应项应该完全相等，而非符号部分已经两两配对配完 n 对了，只剩下符号部分需要检验，而之前的数学过程就是这种思维实验。这在被看作一次“事后检验”的同时，也可以被看作是一次“事前先验”，这样的话，选 n-1 个行做 Laplace 展开，就将成为定理开创者通灵的第一步阶梯：“嗯，两次

检验都正确，这多半是另一条 8th 正确的游戏规则”。【同样的道理，你也可以比较比较选 2 行和选 n-2 行展开的异同，这与“1←VS→n-1”差异不大。】

这些都说明了，行列式以及这之后的，整本线性代数，或许不是人创造的，而是上天借天选之人的手创造的；虽然这是我们写下的，但我们仍然坚称，我们只是发现者，因为这些东西似乎注定地，必然存在于每个高等文明的历史史册中，而人类文明只是其中的一个，有幸能有幸有几个少数村民能以这种方式记录下生命和自己的对话的人；能作为少数发现者存在，这些老先生们在火光中已经满满幸福地看到了自己的子孙们似乎也恍惚着当年自己正恍惚着的东西。

例 1.4 如何利用 Laplace 展开定理简化运算：

对于$\begin{vmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 0 & -1 & 0 & 4 \\ 0 & 1 & 0 & 3 \end{vmatrix}$，你打算用哪一行 or 哪两行展开？运用 Laplace 展开定理，按 3、4 行展开，计算量是最小的：$\sum_{i=1}^{C_4^2} N_iA_i=\begin{vmatrix} -1 & 4 \\ 1 & 3 \end{vmatrix}\times(-1)^{(3+4)+(2+4)}\begin{vmatrix} 1 & 3 \\ 5 & 7 \end{vmatrix}$=-56。

当然，若为了体现 N 与 M 的平权，记$\sum_{i=1}^{C_4^2} N_iA_i$为$\sum_{i=1}^{C_4^2}(-1)^{(i_1+i_2)+(j_1+j_2)}\cdot N_iM_i$也行，但由于$(-1)^{(i_1+i_2)+(j_1+j_2)}$的指数信息是从$N_i$中来的，而这又是继承了$\sum_{i=1}^{n} a_iA_i$的传统，所以从这方面来看，公式这么写的理由占优一些；即使 N、M 俩字母在字母表上挨着的，即前辈们也是考虑过我们的感受的，才决定让公式长这样的。

例 1.5 利用 Laplace 展开定理+性质 7(swap)简算一类行列式：

1.$\begin{vmatrix} \mathbf{A}_{n\times n} & \#_{n\times m} \\ \mathbf{0}_{m\times n} & \mathbf{B}_{m\times m} \end{vmatrix}$、$\begin{vmatrix} \mathbf{A}_{n\times n} & \mathbf{0}_{n\times m} \\ \#_{m\times n} & \mathbf{B}_{m\times m} \end{vmatrix}=(-1)^{n(n+1)}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|=|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|$，即它们的值，均为$|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|$。【同理$\begin{vmatrix} \mathbf{B}_{m\times m} & \#_{m\times n} \\ \mathbf{0}_{n\times m} & \mathbf{A}_{n\times n} \end{vmatrix}$、$\begin{vmatrix} \mathbf{B}_{m\times m} & \mathbf{0}_{m\times n} \\ \#_{n\times m} & \mathbf{A}_{n\times n} \end{vmatrix}$】

其中的$\mathbf{A}_{n\times n}$不像$\sum_{i=1}^{C_4^2} N_iA_i$中的A，A是个带有个前缀的行列式，而$\mathbf{A}_{n\times n}$却是个矩阵——即“行列式符号| |操作的对象”，即一个二维数组、一个有 n 行 n 列的数表。而$|\mathbf{A}_{n\times n}|$才在地位上与 A=$(-1)^{(i_1+i_2+\cdots+i_k)+(j_1+j_2+\cdots+j_k)}\cdot$ M中的M等价，是个行列式。而$\#_{n\times m}$、$\mathbf{0}_{m\times n}$分别表示“各个元素随机取值的 n 行 m 列的数表”、“各元素全是 0 的 m 行 n 列的

数表"。

2.一方面，利用 Laplace：$\begin{vmatrix}\mathbf{0}_{n\times m} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times m} & \#_{m\times n}\end{vmatrix}=(-1)^{\frac{n(n+1)}{2}+\frac{(m+1+m+n)\cdot n}{2}}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|=(-1)^{\frac{n(2m+1+2n+1)}{2}}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|=(-1)^{n(m+n+1)}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|=(-1)^{nm}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|$。

另一方面，我们可以更简单地得到这个结论：$\begin{vmatrix}\mathbf{0}_{n\times m} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times m} & \#_{m\times n}\end{vmatrix}=\begin{vmatrix}\mathbf{0}_{n\times(m-1)} & \mathbf{0}_{n\times 1} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times(m-1)} & \mathbf{B}_{m\times 1} & \#_{m\times n}\end{vmatrix}=(-1)^n\begin{vmatrix}\mathbf{0}_{n\times(m-1)} & \mathbf{A}_{n\times n} & \mathbf{0}_{n\times 1}\\ \mathbf{B}_{m\times(m-1)} & \#_{m\times n} & \mathbf{B}_{m\times 1}\end{vmatrix}=(-1)^{2n}\begin{vmatrix}\mathbf{0}_{n\times(m-2)} & \mathbf{A}_{n\times n} & \mathbf{0}_{n\times 2}\\ \mathbf{B}_{m\times(m-2)} & \#_{m\times n} & \mathbf{B}_{m\times 2}\end{vmatrix}=\ldots=(-1)^{mn}\begin{vmatrix}\mathbf{A}_{n\times n} & \mathbf{0}_{n\times m}\\ \#_{m\times n} & \mathbf{B}_{m\times m}\end{vmatrix}=(-1)^{mn}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|$。

也就是说，先将$\begin{matrix}\mathbf{0}_{n\times m}\\ \mathbf{B}_{m\times m}\end{matrix}$的最后一列做 n 次"与相邻的右列交换位置"，根据性质 7，行列式共改变符号 n 次，并且$\begin{matrix}\mathbf{0}_{n\times m}\\ \mathbf{B}_{m\times m}\end{matrix}$的最后一列，跑到了$\begin{matrix}\mathbf{A}_{n\times n}\\ \#_{m\times n}\end{matrix}$的最后一列上去，而$\begin{matrix}\mathbf{A}_{n\times n}\\ \#_{m\times n}\end{matrix}$整体被向左平移了一格，夹在$\begin{matrix}\mathbf{0}_{n\times(m-1)}\\ \mathbf{B}_{m\times(m-1)}\end{matrix}$与$\begin{matrix}\mathbf{0}_{n\times 1}\\ \mathbf{B}_{m\times 1}\end{matrix}$之间。同样的道理，继续像这样转移$\begin{matrix}\mathbf{0}_{n\times(m-1)}\\ \mathbf{B}_{m\times(m-1)}\end{matrix}$的最后一列到$\begin{matrix}\mathbf{A}_{n\times n}\\ \#_{m\times n}\end{matrix}$的最后一列上去…。总共做了 m×n 次 swap，并且将$\begin{vmatrix}\mathbf{0}_{n\times m} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times m} & \#_{m\times n}\end{vmatrix}$转变为了熟悉的模型：$\begin{vmatrix}\mathbf{A}_{n\times n} & \mathbf{0}_{n\times m}\\ \#_{m\times n} & \mathbf{B}_{m\times m}\end{vmatrix}$。——同理$\begin{vmatrix}\mathbf{0}_{n\times m} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times m} & \#_{m\times n}\end{vmatrix}$的其他三种变体。

因此，对于"次对角线是俩方块阵、主对角线方向含一个矩形0阵"的行列式，我们可直接使用$\begin{vmatrix}\mathbf{0}_{n\times m} & \mathbf{A}_{n\times n}\\ \mathbf{B}_{m\times m} & \#_{m\times n}\end{vmatrix}=(-1)^{mn}\cdot|\mathbf{A}_{n\times n}|\cdot|\mathbf{B}_{m\times m}|$这样的结论；而对于"主对角线是俩方块阵、次对角线方向有一个矩形0阵"的行列式，事情就更简单了。

例 1.6 证明以下乘法公式：

$$\begin{vmatrix}a_{11} & a_{12} & \cdots & a_{1n}\\ a_{21} & a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ a_{n1} & a_{n2} & \cdots & a_{nn}\end{vmatrix}\cdot\begin{vmatrix}b_{11} & b_{12} & \cdots & b_{1n}\\ b_{21} & b_{22} & \cdots & b_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ b_{n1} & b_{n2} & \cdots & b_{nn}\end{vmatrix}=\begin{vmatrix}c_{11} & c_{12} & \cdots & c_{1n}\\ c_{21} & c_{22} & \cdots & c_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ c_{n1} & c_{n2} & \cdots & c_{nn}\end{vmatrix}$$，其中 $c_{ij}=\sum_{k=1}^{n}a_{ik}\cdot b_{kj}$，

(i,j=1,2,3…n)。

根据例 1.5 的结论 1，$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} \cdot \begin{vmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{vmatrix} =$

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} & 0 & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & a_{2n} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & 0 & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 & b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & -1 & \cdots & 0 & b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & b_{n1} & b_{n2} & \cdots & b_{nn} \end{vmatrix} \rightarrow r_i + a_{ij} \cdot r_{n+j}，(i,j=1,2,\ldots,n) =$$

$$\begin{vmatrix} 0 & 0 & \cdots & 0 & c_{11} & c_{12} & \cdots & c_{1n} \\ 0 & 0 & \cdots & 0 & c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & c_{n1} & c_{n2} & \cdots & c_{nn} \\ -1 & 0 & \cdots & 0 & b_{11} & b_{12} & \cdots & b_{1n} \\ 0 & -1 & \cdots & 0 & b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & b_{n1} & b_{n2} & \cdots & b_{nn} \end{vmatrix}$$，再根据例 1.5 的结论 2$=(-1)^{n^2} \cdot \begin{vmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{vmatrix} \cdot$

$$\begin{vmatrix} -1 & 0 & \cdots & 0 \\ 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 \end{vmatrix} = (-1)^{n(n+1)} \cdot \begin{vmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{vmatrix} = \begin{vmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{vmatrix}。$$

这个例子郭老并不推荐在这里出现，他觉得应该讲完矩阵再言及它，这样安排会使得这里出现的乘法原理更自然，直接沿用矩阵的乘法。但我觉得，矩阵的乘法规则的诞生，或许就是从这里受到启发的。并且这也为之后的“方阵的行列式”一小节中，的第三条性质“$|\mathbf{AB}|=|\mathbf{A}|\cdot|\mathbf{B}|$”，提供一部分证明 or 成立的理由：根据这里的$|\mathbf{A}|\cdot|\mathbf{B}|=|\mathbf{C}|$，加上之后的矩阵乘法$\mathbf{A}\cdot\mathbf{B}=\mathbf{C}$所导致的$|\mathbf{A}\cdot\mathbf{B}|=|\mathbf{C}|$，才有了那里的$|\mathbf{A}|\cdot|\mathbf{B}|=|\mathbf{A}\cdot\mathbf{B}|$。

1. 4 行列式的计算

第二节中我们看到，低阶行列式以及对角行列式、三角行列式是很容易计算的。下面我们将利用行列式的性质，将一般阶行列式转换成三角形行列式，或者将高阶行列式转换成低阶行列式来计算。注意结合行列式的具体特点来简化。

例 1.6 从左到右，依次对每列执行以下操作：找 1or 创造 1，将 1 置于主对角线上，用 1 消去该 1 所在列的下方元素，最终得到一个上三角形行列式：

$$D_4=\begin{vmatrix}2&-5&1&2\\-3&7&-1&4\\5&-9&2&7\\4&-6&1&2\end{vmatrix}\xrightarrow{c_1\leftrightarrow c_3}-\begin{vmatrix}1&-5&2&2\\-1&7&-3&4\\2&-9&5&7\\1&-6&4&2\end{vmatrix}\xrightarrow[r_4-r_1]{\substack{r_2+r_1\\r_3-2\cdot r_1}}-$$

$$\begin{vmatrix}1&-5&2&2\\0&2&-1&6\\0&1&1&3\\0&-1&2&0\end{vmatrix}\xrightarrow{r_2\leftrightarrow r_3}\begin{vmatrix}1&-5&2&2\\0&1&1&3\\0&2&-1&6\\0&-1&2&0\end{vmatrix}\xrightarrow[r_4+r_2]{r_3-2\cdot r_2}\begin{vmatrix}1&-5&2&2\\0&1&1&3\\0&0&-3&0\\0&0&3&3\end{vmatrix}\xrightarrow{r_4+r_3}\begin{vmatrix}1&-5&2&2\\0&1&1&3\\0&0&-3&0\\0&0&0&3\end{vmatrix}=-9。$$

我们那么喜欢 1 和-1，是因为用它们能很轻松地消去同一列的下方元素；一般在计算一个没啥规律可言的行列式时，第一步便是看看，在整个行列式共n^2个元素中，有没有 1，若有，则通过行、列互换将其放到(1,1)的位置，同时留意改变行列式的正负。若没有，则要么“找互质的两个数来创造 1”(一般各元素都是整数)：

$\begin{vmatrix}3&8&11\\18&5&-23\\5&76&17\end{vmatrix}\xrightarrow{r_1\times 2}\frac{1}{2}\begin{vmatrix}6&16&22\\18&5&-23\\5&76&17\end{vmatrix}\xrightarrow{r_1-r_3}\frac{1}{2}\begin{vmatrix}1&-60&5\\18&5&-23\\5&76&17\end{vmatrix}$，或者，更通用地：只需保证$a_{11}$≠0(通过 swap 行来保证；一定能保证，否则该行列式值=0)，并进行$r_i-\frac{a_{i1}}{a_{11}}\cdot r_1$，i=2,3,...n 的操作。操作完毕后，第一列的a_{11}下方各元素全是 0 了。之后我们就不要再进行任何涉及第一行 or 第一列的操作了。

再看看包括a_{22}在内的，其右下方↘的共$(n-1)^2$个元素中，有 1 没有，有的话又将其 swap 到a_{22}的位置上去。没有的话，就 swap 行来保证a_{22}≠0(一定能保证，否则该行列式值=0)，再进行$r_i-\frac{a_{i2}}{a_{22}}\cdot r_2$，i=3,...n 的操作。同样，在此之后不再进行任何有关第二行 or 第二列的操作。

以此类推，直到创造了一个上三角形行列式。这是人工做法。而若落实到机器做法——全主元消去法上去，我们将以上操作中的 search 对象“1”改变为“绝对值最大的元素”，仍然是第 k 次则对应在左下角的 n-k+1 阶主子式中共$(n-k+1)^2$个元素中找，且均 swap 到(k,k)位置上的a_{kk}中去。——只要这些元素不全为 0，则找到的“绝对值最大的元素”符合a_{kk}≠0，所以不再需要多余的语句来保证a_{kk}≠0 了。

这样的操作目的在于：既尽可能地规避了a_{kk}=0 的情形，又能够使得$r_i-\frac{a_{ik}}{a_{kk}}\cdot r_k$， i=k+1,...,n 中作为除数的$|a_{kk}|$相对来说最大，以至于用它作除数的各项的数量级、舍入误差均达到最小，让最终计算结果在相同情况下最可靠。

然而列主元消去法相对来说更常用，该方法不需要每次都与$(n-k+1)^2$那么多个元素比较，而是只在a_{kk}所在的第 k 列中，a_{kk}及其之下共$n-k+1$个元素中寻找“绝对值最大的元素”。然后仅通过行变换将其交换到主元素a_{kk}的位置上，再进行消元。可见，列主元消去法仅在一个维度上选取主元，并因此仅需做行交换，而全主元消去法则要花费相当多的时间在两个维度上选取主元，且还需记录系数矩阵列交换信息。

由于计算机做乘除法的运算量远远超过做加减法的运算量，所以我们往往只用乘除的次数来估计运算量——在对第 k 列做消去时，相应元素所对应的行中其余元素也在做$r_i-\frac{a_{ik}}{a_{kk}}\cdot r_k$运算，不过各列均用的是只与第 k 列有关的系数：$\frac{a_{ik}}{a_{kk}}$。那么在第 k 次处理$(n-k+1)^2$个元素构成的方阵时，共计算了 n-k 个$\frac{a_{ik}}{a_{kk}}$，i=k+1,...n，即也就做了 n-k 次乘法运算；接着开始消元，从$r_i-\frac{a_{ik}}{a_{kk}}\cdot r_k$(i=k+1,...,n)=$a_{ij}-\frac{a_{ik}}{a_{kk}}\cdot a_{kj}$(i,j=k+1,...,n)中可看出【注：j 也从 k+1 列开始算起，这说明第 k 列的第 k+1 行及以下，不需运算便知道它们是 0】，需做$(n-k)^2$次乘法运算。

那么总共计算量=$\sum_{k=1}^{n-1}[(n-k)^2+n-k]$=$[1^2+2^2+...+(n-1)^2]$+(1+2+...+n-1)，利用级数公式$\sum_{k=1}^{n-1}k^2=\frac{1}{6}n(n+1)(2n+1)$，可得到原式=$\frac{2n-1}{6}n(n-1)+\frac{1}{2}n(n-1)=\frac{n+1}{3}n(n-1)=\frac{1}{3}(n^3-n)$。也就是说，在用列主元消去法操作方阵时，需要进行$\frac{1}{3}(n^3-n)$次乘法运算。——而若要用列主元消去法操作多元一次方程组(增广矩阵)，则还会加上增补列$b_i-\frac{a_{ik}}{a_{kk}}\cdot b_k$(i=k+1,...,n)的 n-k 次乘法运算，所引进的总的$\frac{1}{2}n(n-1)$次；以及回代求解：$x_n=\frac{b_n}{a_{nn}}$、$x_k=\frac{b_k-\sum_{j=k+1}^{n}a_{kj}\cdot x_j}{a_{kk}}$(k=n-1,...,1，注意倒序！)，的过程中共(1+2+...+n)=$\frac{1}{2}n(n+1)$次乘法运算，总共$\frac{1}{3}(n^3-n)+\frac{1}{2}n(n-1)+\frac{1}{2}n(n+1)=\frac{n^3}{3}+n^2-\frac{n}{3}$次乘法运算。

可见列主元消去法操作方阵的乘除运算次数在n^3这个数量级。而之前我们说过，用

定义法来求行列式值需要做(n-1)·n!次乘法运算，而(n-1)·n!>>n^3，因此，列主元消去法所代表的(高斯)主元素消去法，是更广为使用的，不论在人工操作上，还是在计算机程序里。

这些均适用于各元素非整数情况。还可能有其他可行方案，用以优化舍入误差，不过优化的同时也多半意味着计算量的增大，兼顾这两个相互矛盾的因素，列主元消去法已经不错了。

一些特殊的对角对称型行列式的对应求法：

例 1.7 计算$\begin{vmatrix} a & b & \cdots & b \\ b & a & \cdots & b \\ \vdots & \vdots & \ddots & \vdots \\ b & b & \cdots & a \end{vmatrix}_n$(主对角线上各元素相等，主对角线之外的元素也相等)：

我们先举个 4 阶的例子：$\begin{vmatrix} a & b & b & b \\ b & a & b & b \\ b & b & a & b \\ b & b & b & a \end{vmatrix} \xrightarrow{r_1+r_i(i=2,3,4)} \begin{vmatrix} a+3b & a+3b & a+3b & a+3b \\ b & a & b & b \\ b & b & a & b \\ b & b & b & a \end{vmatrix}$ $\xrightarrow{r_1\div(a+3b)} (a+3b)\begin{vmatrix} 1 & 1 & 1 & 1 \\ b & a & b & b \\ b & b & a & b \\ b & b & b & a \end{vmatrix} \xrightarrow{r_i-b\cdot r_1(i=2,3,4)} (a+3b)\begin{vmatrix} 1 & 1 & 1 & 1 \\ 0 & a-b & 0 & 0 \\ 0 & 0 & a-b & 0 \\ 0 & 0 & 0 & a-b \end{vmatrix} = (a+3b)\cdot$ $(a-b)^3$。依葫芦画瓢：将其中的两个 3 改为两个 n-1 即可得到 n 阶情况时的结论：$[a+(n-1)b]\cdot(a-b)^{n-1}$。

例 1.8 设行列式 D=$|a_{ij}|_n$的阶数 n 为奇数，且$a_{ij}=-a_{ji}$(i,j=1,2,...,n)，求 D：

首先，易知主对角线上各元素a_{ii}=0(i=1,2,...,n)，故 D=$\begin{vmatrix} 0 & a_{12} & \cdots & a_{1n} \\ -a_{12} & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{1n} & -a_{2n} & \cdots & 0 \end{vmatrix}_{2m+1}$。

则 D=$\begin{vmatrix} 0 & a_{12} & \cdots & a_{1n} \\ -a_{12} & 0 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -a_{1n} & -a_{2n} & \cdots & 0 \end{vmatrix}_n \xrightarrow{r_i\div(-1)(i=1,2,3,\ldots,n)} (-1)^n\begin{vmatrix} 0 & -a_{12} & \cdots & -a_{1n} \\ a_{12} & 0 & \cdots & -a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & 0 \end{vmatrix}_n = (-1)^n D^T$ $=(-1)^n D$。当 n 为奇数时，D=−D，此时 D=0；当 n 为奇数时，这么做得到的是 D=D，又回去了，得不到 D，估计此时只能硬算了——不过在展开D_{2m}时，其展开式中各个子式可能又含有奇数阶主子式，可直接利用结论让这些奇数阶主子式=0。

例 1.9 计算行列式$\begin{vmatrix} a & b & & & \\ b & a & b & & \\ & b & \ddots & \ddots & \\ & & \ddots & a & b \\ & & & b & a \end{vmatrix}_n$：

$D_n \xrightarrow{\text{按第一行展开}} a\cdot M_{11} - b\cdot M_{12}$，其中$M_{11}=D_{n-1}$，$M_{12} \xrightarrow{\text{按第一列展开}} D_{n-2}$。因此$D_n=a\cdot D_{n-1}-b\cdot D_{n-2}$。若取 a=2、b=1，则简化为$D_n=2\cdot D_{n-1}-D_{n-2}$，此时$D_n-D_{n-1}=D_{n-1}-D_{n-2}$，可见这是个等差数列，公差 d=$D_2-D_1$=3-2=1，因此$D_n=D_1+(n-1)d=n+1$。

若不取 a、b 的特定值，求$D_n=a\cdot D_{n-1}-b\cdot D_{n-2}$的通解，其中一个较为简单的办法，需要用到特征向量部分的有关知识。斐波那契数列的通解也是其的一个特例。

****

(这是一条分割线；接下来的三道题组成的 group2 用于引入增行增列法)

例 2.0 计算行列式$\begin{vmatrix} x & a_1 & a_2 & \cdots & a_n \\ a_1 & x & a_2 & \cdots & a_n \\ a_1 & a_2 & x & \cdots & a_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & a_3 & \cdots & x \end{vmatrix}$：

方法一：除最后一行外的上面 n-1 行，都减去第 n 行；从左到右，每一列加上后面所有列；最后按第一列展开：(这种方法没有什么通用规律可言，有点初学者的鲁莽；计算量大，思路有点不清晰)

$$\begin{vmatrix} x & a_1 & a_2 & \cdots & a_n \\ a_1 & x & a_2 & \cdots & a_n \\ a_1 & a_2 & x & \cdots & a_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & a_3 & \cdots & x \end{vmatrix} = \begin{vmatrix} x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & a_n-x \\ 0 & x-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & a_n-x \\ 0 & 0 & x-a_3 & \cdots & a_{n-1}-a_n & a_n-x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & x-a_n & a_n-x \\ a_1 & a_2 & a_3 & \cdots & a_n & x \end{vmatrix} =$$

$$\begin{vmatrix} 0 & a_1-x & a_2-x & \cdots & a_{n-1}-x & a_n-x \\ 0 & 0 & a_2-x & \cdots & a_{n-1}-x & a_n-x \\ 0 & 0 & 0 & \cdots & a_{n-1}-x & a_n-x \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & a_n-x \\ a_1+\cdots+x & a_2+\cdots+x & a_3+\cdots+x & \cdots & a_n+x & x \end{vmatrix} =$$

$$(a_1+\cdots+x)\cdot(-1)^{n+2}\cdot\begin{vmatrix} a_1-x & a_2-x & \cdots & a_{n-1}-x & a_n-x \\ 0 & a_2-x & \cdots & a_{n-1}-x & a_n-x \\ 0 & 0 & \ddots & \vdots & a_n-x \\ \vdots & \vdots & \ddots & a_{n-1}-x & \vdots \\ 0 & 0 & \cdots & 0 & a_n-x \end{vmatrix}=$$

$$(a_1+\cdots+x)\cdot(-1)^{n+2}\cdot(-1)^{n}\cdot[(x-a_1)*\ldots*(x-a_n)]=(a_1+\cdots+x)\cdot[(x-a_1)*\ldots*(x-a_n)]$$

方法二：第一列加上后面所有列，并提出公因子$(a_1+\cdots+x)$；$-a_i\times$第一列，加到从左往右数第 2 列开始的每一列(对应第 i+1 列，i≥1)上，将其变为一个下三角形行列式。

(这种方法适用于一些每行(每列)之和恰好都相同的行列式，提取系数后用以创造一行元素 1，用以消元；计算量小，思路清晰，之后我们会再次用到)

方法三：增行增列法 1(eventually 创造个下三角)：原式=
$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & x & a_1 & a_2 & \cdots & a_n \\ 1 & a_1 & x & a_2 & \cdots & a_n \\ 1 & a_1 & a_2 & x & \cdots & a_n \\ 1 & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_1 & a_2 & a_3 & \cdots & x \end{vmatrix}=$$

$$\begin{vmatrix} 1 & -a_1 & -a_2 & -a_3 & \cdots & -a_n & -a_n \\ 1 & x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & 0 \\ 1 & 0 & x-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & 0 \\ 1 & 0 & 0 & x-a_3 & \vdots & \vdots & 0 \\ 1 & \vdots & \vdots & \vdots & \ddots & a_{n-1}-a_n & \vdots \\ 1 & \vdots & \vdots & \vdots & \vdots & x-a_n & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 & x-a_n \end{vmatrix}=$$

$$\begin{vmatrix} 1+\frac{a_n}{x-a_n} & -a_1 & -a_2 & -a_3 & \cdots & -a_n & 0 \\ 1 & x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & 0 \\ 1 & 0 & x-a_2 & a_2-a_3 & \cdots & a_{n-1}-a_n & 0 \\ 1 & 0 & 0 & x-a_3 & \vdots & \vdots & 0 \\ 1 & \vdots & \vdots & \vdots & \ddots & a_{n-1}-a_n & \vdots \\ 1 & \vdots & \vdots & \vdots & \vdots & x-a_n & 0 \\ 1 & 0 & 0 & 0 & \cdots & 0 & x-a_n \end{vmatrix}=(x-$$

$$a_n)\cdot\begin{vmatrix} 1+\frac{a_n}{x-a_n}+\frac{a_n}{x-a_n} & -a_1 & -a_2 & -a_3 & \cdots & 0 \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & 0 \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & 0 & x-a_2 & a_2-a_3 & \cdots & 0 \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & 0 & 0 & x-a_3 & \vdots & \vdots \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & \vdots & \vdots & \vdots & \ddots & 0 \\ 1 & \vdots & \vdots & \vdots & \vdots & x-a_n \end{vmatrix}=(x-a_n)\cdot(x-$$

$$a_n)\cdot\begin{vmatrix} 1+\frac{a_n}{x-a_n}+\frac{a_n}{x-a_n} & -a_1 & -a_2 & -a_3 & \cdots & -a_{n-1} \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & a_{n-2}-a_{n-1} \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & 0 & x-a_2 & a_2-a_3 & \cdots & a_{n-2}-a_{n-1} \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & 0 & 0 & x-a_3 & \vdots & \vdots \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & \vdots & \vdots & \vdots & \ddots & a_{n-2}-a_{n-1} \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & \vdots & \vdots & \vdots & \vdots & x-a_{n-1} \end{vmatrix}=(x-a_n)\cdot(x-$$

$$a_n)\cdot\begin{vmatrix} 1+\frac{a_n}{x-a_n}+\frac{a_n}{x-a_n}+\left(1-\frac{a_{n-1}-a_n}{x-a_n}\right)\cdot\frac{a_{n-1}}{x-a_{n-1}} & -a_1 & -a_2 & -a_3 & \cdots & 0 \\ (1-\frac{a_{n-1}-a_n}{x-a_n})(1-\frac{a_{n-2}-a_{n-1}}{x-a_{n-1}}) & x-a_1 & a_1-a_2 & a_2-a_3 & \cdots & 0 \\ (1-\frac{a_{n-1}-a_n}{x-a_n})(1-\frac{a_{n-2}-a_{n-1}}{x-a_{n-1}}) & 0 & x-a_2 & a_2-a_3 & \cdots & 0 \\ (1-\frac{a_{n-1}-a_n}{x-a_n})(1-\frac{a_{n-2}-a_{n-1}}{x-a_{n-1}}) & 0 & 0 & x-a_3 & \vdots & \vdots \\ (1-\frac{a_{n-1}-a_n}{x-a_n})(1-\frac{a_{n-2}-a_{n-1}}{x-a_{n-1}}) & \vdots & \vdots & \vdots & \ddots & 0 \\ 1-\frac{a_{n-1}-a_n}{x-a_n} & \vdots & \vdots & \vdots & \vdots & x-a_{n-1} \end{vmatrix}=[(x-a_1)\cdot(x-$$

$$a_2)\ldots(x-a_n)\cdot(x-a_n)]\cdot[1+\frac{a_n}{x-a_n}+\frac{a_n}{x-a_n}+\left(1-\frac{a_{n-1}-a_n}{x-a_n}\right)\cdot\frac{a_{n-1}}{x-a_{n-1}}+(1-\frac{a_{n-1}-a_n}{x-a_n})(1-\frac{a_{n-2}-a_{n-1}}{x-a_{n-1}})\cdot\frac{a_{n-2}}{x-a_{n-2}}+\cdots+]=(\frac{x-a_n}{x-a_n}+\frac{a_n}{x-a_n}+\frac{a_n}{x-a_n}+\frac{a_{n-1}}{x-a_n}\ldots+\frac{a_1}{x-a_1})\cdot[(x-a_1)\cdot(x-a_2)\ldots(x-a_n)\cdot(x-a_n)]=(a_1+\cdots+x)\cdot[(x-a_1)*\ldots*(x-a_n)]$$

当然，你也可以这样来操作它：增行增列法 2(也是 eventually 创造个下三角)：

$$\begin{vmatrix} 1 & -a_1 & -a_1 & -a_2 & \cdots & -a_{n-1} & -a_n \\ 1 & x-a_1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & x-a_1 & 0 & \cdots & \vdots & \vdots \\ 1 & 0 & a_2-a_1 & x-a_3 & \vdots & \vdots & \vdots \\ 1 & \vdots & a_2-a_1 & \vdots & \ddots & 0 & 0 \\ 1 & \vdots & a_2-a_1 & \vdots & \vdots & x-a_{n-1} & 0 \\ 1 & 0 & a_2-a_1 & 0 & \cdots & a_n-a_{n-1} & x-a_n \end{vmatrix}$$

，同样要将第一行除了 1 之后的那些棋子消掉，可见第二行和第三行是可以直接消去第一行中的对应列的元素$-a_1$和$-a_1$的，不过你不能用第三行及以后的行直接消去第一行的元素，因为你在用(4,4)的元素去消去(1,4)

的元素$-a_2$时，你左边即(4,3)的元素不是 0，会干扰已经消空了的(1,3)位置上的 0；因此你在用(4,4)去让(1,4)归零时，注意保证第四行除了(4,1)、(4,4)之外的元素全为 0；因此你需要在之前用(3,3)归零(1,3)的同时，让(3,3)也将(4,3)归零；同样的道理，你也得用(3,3)将(4,3)之下的其他(≥4,3)们归零，以除后患。

这便是增行增列法的精髓：将一切行变换信息记录在左边的一列 1 的运算当中，并最终接力给(1,1)位置处的 1，然后创造个下三角行列式，输出主对角元素乘积作为答案。或许对这个例子用增行增列法并不恰当，我们会给出一个更恰当的例子让你体会体会，什么时候用什么方法更好。

例 2.1 计算行列式$\begin{vmatrix} x & a_2 & a_3 & \cdots & a_n \\ a_1 & x & a_3 & \cdots & a_n \\ a_1 & a_2 & x & \cdots & a_n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & a_3 & \cdots & x \end{vmatrix}$:

乍看上去是不是和之前那道题一模一样？可惜事实上上面那道题是个 n+1 阶行列式，而这道题是个 n 阶行列式。仔细看你才能发现其中的端倪。

这道题就比较适合使用增行增列法：

$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & x & a_2 & a_3 & \cdots & a_n \\ 1 & a_1 & x & a_3 & \cdots & a_n \\ 1 & a_1 & a_2 & x & \cdots & a_n \\ 1 & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_1 & a_2 & a_3 & \cdots & x \end{vmatrix} = \begin{vmatrix} 1 & -a_1 & -a_2 & -a_3 & \cdots & -a_n \\ 1 & x-a_1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & x-a_2 & 0 & \cdots & 0 \\ 1 & 0 & 0 & x-a_3 & \cdots & 0 \\ 1 & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & x-a_n \end{vmatrix} =$$

$$\begin{vmatrix} 1 & -a_1 & -a_2 & -a_3 & \cdots & -a_n \\ 1 & x-a_1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & x-a_2 & 0 & \cdots & 0 \\ 1 & 0 & 0 & x-a_3 & \cdots & 0 \\ 1 & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & x-a_n \end{vmatrix} = \begin{vmatrix} 1+\frac{a_1}{x-a_1}\ldots+\frac{a_n}{x-a_n} & 0 & 0 & 0 & \cdots & 0 \\ 1 & x-a_1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & x-a_2 & 0 & \cdots & 0 \\ 1 & 0 & 0 & x-a_3 & \cdots & 0 \\ 1 & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & x-a_n \end{vmatrix} = (1+\frac{a_1}{x-a_1}\ldots+\frac{a_n}{x-a_n})\cdot[(x-a_1)\cdot(x-a_2)\ldots(x-a_n)]$$

例 2.2 计算行列式 $\begin{vmatrix} a+x & a & a & \cdots & a \\ a & a+x & a & \cdots & a \\ a & a & a+x & \cdots & a \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a & a & a & \cdots & a+x \end{vmatrix}$:

暂时取 a=1，其他值类似；直接用增行增列法：

$$D_n=\begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1+x_n & 1 & \cdots & 1 \\ 1 & 1 & 1+x_{n-1} & \cdots & 1 \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1+x_1 \end{vmatrix}=\begin{vmatrix} 1 & -1 & -1 & -1 & -1 \\ 1 & x_n & 0 & \cdots & 0 \\ 1 & 0 & x_{n-1} & \cdots & 0 \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & x_1 \end{vmatrix}=\begin{vmatrix} 1+\sum_{j=1}^{n}\frac{1}{x_j} & 0 & 0 & 0 & 0 \\ 1 & x_n & 0 & \cdots & 0 \\ 1 & 0 & x_{n-1} & \cdots & 0 \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & x_1 \end{vmatrix}=\prod_{i=1}^{n}x_i\cdot\left(1+\sum_{j=1}^{n}\frac{1}{x_j}\right)$$

当 a 保留字母形式时，答案变为$\prod_{i=1}^{n}x_i\cdot\left(1+\sum_{j=1}^{n}\frac{a}{x_j}\right)$；我当时做这道题的时候，还想了个无可奈何的做法，并觉得这方法除了我之外没人想得到，因为它看上去太麻烦了。可惜由于办法较为麻烦而并不引以为豪- -，所以该方法可看可不看，它有点尝试着沿袭了之前递推的套路，这需要一双好眼睛找到通项：

$$D_n=\begin{vmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1+x_{n-1} & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1+x_1 \end{vmatrix}+\begin{vmatrix} x_n & 0 & \cdots & 0 \\ 1 & 1+x_{n-1} & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1+x_1 \end{vmatrix}=\begin{vmatrix} 1 & 1 & \cdots & 1 \\ 0 & x_{n-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_1 \end{vmatrix}+x_n\begin{vmatrix} 1+x_{n-1} & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1+x_1 \end{vmatrix}=$$

$$\begin{vmatrix} x_{n-1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & x_1 \end{vmatrix}+x_nD_{n-1}=\prod_{i=1}^{n-1}x_i+x_nD_{n-1}=\prod_{i=1}^{n-1}x_i+x_n\left(\prod_{i=1}^{n-2}x_i+x_{n-1}D_{n-2}\right)=\prod_{i=1}^{n-1}x_i+x_n\prod_{i=1}^{n-2}x_i$$

$$+x_nx_{n-1}D_{n-2}=\prod_{i=1}^{n-1}x_i+\prod_{i=n}^{n}x_i\prod_{i=1}^{n-2}x_i+\prod_{i=n-1}^{n}x_i\left(\prod_{i=1}^{n-3}x_i+x_{n-2}D_{n-3}\right)=$$

$$\prod_{i=1}^{n-1}x_i+\prod_{i=n}^{n}x_i\prod_{i=1}^{n-2}x_i+\prod_{i=n-1}^{n}x_i\prod_{i=1}^{n-3}x_i+\prod_{i=n-2}^{n}x_i\,D_{n-3}=\prod_{i=1}^{n}x_i\cdot\left(1+\sum_{j=1}^{n}\frac{1}{x_j}\right)。$$

这是第二个“反面教材”，从“方法选得不适”的角度。不过从“一题多解”的角度，它又是个“正面教材”。

****

一些另类有趣的行列式及对应求法：

例 2.3 计算行列式$\begin{vmatrix} x & -1 & 0 & \cdots & 0 & 0 \\ 0 & x & -1 & \cdots & 0 & 0 \\ 0 & 0 & x & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & x & -1 \\ a_n & a_{n-1} & a_{n-2} & \cdots & a_2 & x+a_1 \end{vmatrix}$：

这个行列式长相不像以前的那 4+3=7 个例子那么对称，不怎么好看；之前的可能是“美得舍不得下手”(完美对称而找不到切入点)，这个估计是“没兴趣对它动手”(没规律可言而找不到切入点)，反正不会做就对了，是么- -。其实这个行列式算出结果是简单的，然而它的价值在于，若将问题换为：考虑将其结果这个多项式，写成一个行列式，要怎么写：

$\begin{vmatrix} x & -1 & 0 & \cdots & 0 & 0 \\ 0 & x & -1 & \cdots & 0 & 0 \\ 0 & 0 & x & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & x & -1 \\ a_n & a_{n-1} & a_{n-2} & \cdots & a_2 & x+a_1 \end{vmatrix} \xrightarrow[(i=n,n-1,\dots 2)]{c_{i-1}+x\cdot c_i} x^n + a_1x^{n-1} + \cdots + a_n$，如你所见，左到右，它只是个题；而右到左，它却变成了一个小的研究性问题：如何将一个 n 次多项式写成(还原成)矩阵的形式。

例 2.4 计算行列式$\begin{vmatrix} a+b & ab & 0 & \cdots & 0 & 0 & 0 \\ 1 & a+b & ab & \cdots & 0 & 0 & 0 \\ 0 & 1 & a+b & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a+b & ab & 0 \\ 0 & 0 & 0 & \cdots & 1 & a+b & ab \\ 0 & 0 & 0 & \cdots & 0 & 1 & a+b \end{vmatrix}_n$：

方法一：递推：

$$\begin{vmatrix} a+b & ab & 0 & \cdots & 0 & 0 & 0 \\ 1 & a+b & ab & \cdots & 0 & 0 & 0 \\ 0 & 1 & a+b & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a+b & ab & 0 \\ 0 & 0 & 0 & \cdots & 1 & a+b & ab \\ 0 & 0 & 0 & \cdots & 0 & 1 & a+b \end{vmatrix} \xrightarrow[(i=2,3,\ldots n)]{c_i+(-a)\cdot c_{i-1}} \begin{vmatrix} a+b-a^2 & a^3 & \cdots & \cdots & \cdots & (-1)^{n-1}a^n \\ 1 & b & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & b & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & b & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & b & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & b \end{vmatrix} =$$

$$(a+b)\cdot b^{n-1} - \begin{vmatrix} -a^2 & a^3 & \cdots & \cdots & \cdots & (-1)^{n-1}a^n \\ 1 & b & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & b & 0 & 0 \\ 0 & 0 & \cdots & 1 & b & 0 \\ 0 & 0 & \cdots & 0 & 1 & b \end{vmatrix} = (a+b)\cdot b^{n-1} - [(-a^2)\cdot b^{n-2} -$$

$$\begin{vmatrix} a^3 & \cdots & \cdots & \cdots & (-1)^{n-1}a^n \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & b & 0 & 0 \\ 0 & \cdots & 1 & b & 0 \\ 0 & \cdots & 0 & 1 & b \end{vmatrix}] = (a+b)\cdot b^{n-1} - (-a^2)\cdot b^{n-2} + a^3\cdot b^{n-3} - (-a^4)\cdot b^{n-4}\cdots$$

$=b^n + a\cdot b^{n-1} + a^2\cdot b^{n-2} + a^3\cdot b^{n-3} + a^4\cdot b^{n-4}\cdots + a^n = \frac{b^{n+1}-a^{n+1}}{b-a}(b\neq a)$或$(n+1)a^n(b=a)$

方法二：解方程组：

$$D_n = \begin{vmatrix} a+b & ab & 0 & \cdots & 0 & 0 & 0 \\ 1 & a+b & ab & \cdots & 0 & 0 & 0 \\ 0 & 1 & a+b & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a+b & ab & 0 \\ 0 & 0 & 0 & \cdots & 1 & a+b & ab \\ 0 & 0 & 0 & \cdots & 0 & 1 & a+b \end{vmatrix}_n = (a+b)\cdot D_{n-1} - ab\cdot D_{n-2}，$$

到此为止，不再向下展开，看看我们可以弄到手什么东西：对“$D_n=(a+b)\cdot D_{n-1}-ab\cdot D_{n-2}$”进行两类移项操作，分别得到形式上相同的两个方程，组成一个方程组：$D_n - a\cdot D_{n-1} = b\cdot(D_{n-1}-a\cdot D_{n-2})$；$D_n - b\cdot D_{n-1} = a\cdot(D_{n-1}-b\cdot D_{n-2})$。

对其中一个方程向下级拓展：$D_n - a\cdot D_{n-1} = b\cdot(D_{n-1}-a\cdot D_{n-2}) = b^2\cdot(D_{n-2}-a\cdot D_{n-3}) = \ldots = b^{n-2}\cdot(D_2 - a\cdot D_1) = b^n$。类似地，另一个方程直接得$D_n - b\cdot D_{n-1} = a\cdot$

$(D_{n-1}-b\cdot D_{n-2})=\ldots=a^n$。于是我们得到了一个二元线性方程组：$D_n-a\cdot D_{n-1}=b^n$、$D_n-b\cdot D_{n-1}=a^n$。利用克莱姆法则，解得：当$\begin{vmatrix}1 & -a\\ 1 & -b\end{vmatrix}\neq 0$ 时，$D_n=\dfrac{\begin{vmatrix}b^n & -a\\ a^n & -b\end{vmatrix}}{\begin{vmatrix}1 & -a\\ 1 & -b\end{vmatrix}}=\dfrac{b^{n+1}-a^{n+1}}{b-a}$；

当$\begin{vmatrix}1 & -a\\ 1 & -b\end{vmatrix}=0$ 时，$D_n=a^n+a\cdot D_{n-1}=a^n+a\cdot(a^{n-1}+a\cdot D_{n-2})=2\cdot a^n+a^2\cdot D_{n-2}=\ldots=(n-2)\cdot a^n+a^{n-2}\cdot D_2=(n-2)\cdot a^n+a^{n-2}\cdot(a^2+a\cdot D_1)$，而$D_1=2a$，因此原式

$=(n+1)a^n$。

例 2.5 计算行列式$\begin{vmatrix}1 & 2 & 3 & \cdots & n\\ 2 & 3 & 4 & \cdots & 1\\ 3 & 4 & 5 & \cdots & 2\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ n & 1 & 2 & \cdots & n-1\end{vmatrix}$；

方法一：其余所有行，加到第一行。

$$\begin{vmatrix}1 & 2 & 3 & \cdots & n\\ 2 & 3 & 4 & \cdots & 1\\ 3 & 4 & 5 & \cdots & 2\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ n & 1 & 2 & \cdots & n-1\end{vmatrix}=\frac{(1+n)n}{2}\begin{vmatrix}1 & 1 & 1 & \cdots & 1\\ 2 & 3 & 4 & \cdots & 1\\ 3 & 4 & 5 & \cdots & 2\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ n & 1 & 2 & \cdots & n-1\end{vmatrix}=\text{第 i 行减 i-1 个第一行}$$

$$\frac{(1+n)n}{2}\begin{vmatrix}1 & 1 & 1 & \cdots & 1 & 1 & 1\\ 1 & 2 & 3 & \cdots & n-2 & n-1 & 0\\ 1 & 2 & 3 & \cdots & n-2 & -1 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots\\ 1 & 2 & 3 & \cdots & -2 & -1 & 0\\ 1 & 2 & 3-n & \cdots & -2 & -1 & 0\\ 1 & 2-n & 3-n & \cdots & -2 & -1 & 0\end{vmatrix}=\frac{(1+n)n}{2}\begin{vmatrix}1 & 2 & 3 & \cdots & n-2 & n-1\\ 1 & 2 & 3 & \cdots & n-2 & -1\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 1 & 2 & 3 & \cdots & -2 & -1\\ 1 & 2 & 3-n & \cdots & -2 & -1\\ 1 & 2-n & 3-n & \cdots & -2 & -1\end{vmatrix}(-1)^{1+n}$$

$$=(-1)^{1+n}\cdot\frac{(1+n)n}{2}\begin{vmatrix}1 & 0 & 0 & \cdots & 0 & 0\\ 1 & 0 & 0 & \cdots & 0 & -n\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 1 & 0 & 0 & \cdots & -n & -n\\ 1 & 0 & -n & \cdots & -n & -n\\ 1 & -n & -n & \cdots & -n & -n\end{vmatrix}=(-1)^{1+n}\cdot\frac{(1+n)n}{2}\begin{vmatrix}0 & 0 & \cdots & 0 & -n\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & \cdots & -n & -n\\ 0 & -n & \cdots & -n & -n\\ -n & -n & \cdots & -n & -n\end{vmatrix}$$

$$=(-1)^{1+n}\cdot\frac{(1+n)n}{2}\cdot(-n)^{n-2}\cdot(-1)^{\frac{(n-2)(n-3)}{2}}=(-1)^{2n-1}\cdot\frac{(1+n)}{2}\cdot n^{n-1}\cdot-(-1)^{\frac{n(n-1)}{2}}$$

$$=\frac{(1+n)}{2}\cdot n^{n-1}\cdot(-1)^{\frac{n(n-1)}{2}}$$

方法二：从下往上，下行减上行。

$$\begin{vmatrix} 1 & 2 & 3 & \cdots & n \\ 2 & 3 & 4 & \cdots & 1 \\ 3 & 4 & 5 & \cdots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ n & 1 & 2 & \cdots & n-1 \end{vmatrix} = \begin{vmatrix} 1 & 2 & 3 & \cdots & n-2 & n-1 & n \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1-n \\ 1 & 1 & 1 & \cdots & 1 & 1-n & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1 & 1-n & \cdots & 1 & 1 & 1 \\ 1 & 1-n & 1 & \cdots & 1 & 1 & 1 \end{vmatrix} =$$

$$\begin{vmatrix} \frac{(1+n)n}{2} & 2 & 3 & \cdots & n-2 & n-1 & n \\ 0 & 1 & 1 & \cdots & 1 & 1 & 1-n \\ 0 & 1 & 1 & \cdots & 1 & 1-n & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1-n & \cdots & 1 & 1 & 1 \\ 0 & 1-n & 1 & \cdots & 1 & 1 & 1 \end{vmatrix} = \frac{(1+n)n}{2} \begin{vmatrix} 1 & 1 & \cdots & 1 & 1 & 1-n \\ 1 & 1 & \cdots & 1 & 1-n & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1-n & \cdots & 1 & 1 & 1 \\ 1-n & 1 & \cdots & 1 & 1 & 1 \end{vmatrix} =$$

$$\frac{(1+n)n}{2} \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1-n \\ 1 & 1 & 1 & \cdots & 1 & 1-n & 1 \\ 1 & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & \cdots & 1 & 1 & 1 \\ 1 & 1 & 1-n & \cdots & 1 & 1 & 1 \\ 1 & 1-n & 1 & \cdots & 1 & 1 & 1 \end{vmatrix} = \frac{(1+n)n}{2} \begin{vmatrix} 1 & -1 & -1 & \cdots & -1 & -1 & -1 \\ 1 & 0 & 0 & \cdots & 0 & 0 & -n \\ 1 & 0 & 0 & \cdots & 0 & -n & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & -n & \cdots & 0 & 0 & 0 \\ 1 & -n & 0 & \cdots & 0 & 0 & 0 \end{vmatrix} =$$

$$\frac{(1+n)n}{2} \begin{vmatrix} 1+\left(\frac{1}{-n}\right)^{n-1} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & 0 & \cdots & 0 & 0 & -n \\ 1 & 0 & 0 & \cdots & 0 & -n & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 0 & -n & \cdots & 0 & 0 & 0 \\ 1 & -n & 0 & \cdots & 0 & 0 & 0 \end{vmatrix} = \frac{(1+n)n}{2} \begin{vmatrix} 0 & 0 & \cdots & 0 & 0 & -n \\ 0 & 0 & \cdots & 0 & -n & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -n & \cdots & 0 & 0 & 0 \\ -n & 0 & \cdots & 0 & 0 & 0 \end{vmatrix}.$$

$$\left(1+(n-1)\frac{1}{-n}\right) = \frac{(1+n)}{2}(-n)^{n-1}\cdot(-1)^{\frac{(n-1)(n-2)}{2}} = \frac{(1+n)}{2}n^{n-1}\cdot(-1)^{\frac{(n-1)(n-2)}{2}+n-1}$$

$$= \frac{(1+n)}{2}n^{n-1}\cdot(-1)^{\frac{(n-1)(n-2+2)}{2}} = \frac{(1+n)}{2}n^{n-1}\cdot(-1)^{\frac{n(n-1)}{2}}$$

****

例 2.6 范得蒙德行列式：

$$D_n=\begin{vmatrix}1&1&1&\cdots&1\\a_1&a_2&a_3&\cdots&a_n\\a_1^2&a_2^2&a_3^2&\cdots&a_n^2\\\vdots&\vdots&\vdots&\ddots&\vdots\\a_1^{n-1}&a_2^{n-1}&a_3^{n-1}&\cdots&a_n^{n-1}\end{vmatrix}=\prod_{1\le i<j\le n}(a_j-a_i)。$$

仍然对阶数 n 运用数学归纳法：当 n=2 时，$D_2=\begin{vmatrix}1&1\\a_1&a_2\end{vmatrix}$。而由于：

$$D_n\xrightarrow[(i=n,n-1,\dots2)]{r_i-a_1\cdot r_{i-1}}\begin{vmatrix}1&1&1&\cdots&1\\0&a_2-a_1&a_3-a_1&\cdots&a_n-a_1\\0&a_2(a_2-a_1)&a_3(a_3-a_1)&\cdots&a_n(a_n-a_1)\\\vdots&\vdots&\vdots&\ddots&\vdots\\0&a_2^{n-2}(a_2-a_1)&a_3^{n-2}(a_3-a_1)&\cdots&a_n^{n-2}(a_n-a_1)\end{vmatrix}=$$

$$\begin{vmatrix}a_2-a_1&a_3-a_1&\cdots&a_n-a_1\\a_2(a_2-a_1)&a_3(a_3-a_1)&\cdots&a_n(a_n-a_1)\\\vdots&\vdots&\ddots&\vdots\\a_2^{n-2}(a_2-a_1)&a_3^{n-2}(a_3-a_1)&\cdots&a_n^{n-2}(a_n-a_1)\end{vmatrix}=(a_2-a_1)(a_3-a_1)\cdots(a_n-a_1)\begin{vmatrix}1&1&\cdots&1\\a_2&a_3&\cdots&a_n\\\vdots&\vdots&\ddots&\vdots\\a_2^{n-2}&a_3^{n-2}&\cdots&a_n^{n-2}\end{vmatrix}$$

$=(a_2-a_1)(a_3-a_1)\cdots(a_n-a_1)D_{n-1}$，假设结论对 n-1 阶成立，则

$D_n=(a_2-a_1)(a_3-a_1)\cdots(a_n-a_1)\prod_{n\ge j>i\ge 2}(a_j-a_i)=\prod_{j=2}^{n}a_j-a_1\cdot\prod_{i=2}^{n-1}\prod_{j=i+1}^{n}a_j-a_i=\prod_{i=1}^{n-1}\prod_{j=i+1}^{n}a_j-a_i=\prod_{n\ge j>i\ge 1}(a_j-a_i)$，即结论对$D_n$也成立。

【注：这里的$D_{n-1}\ne\prod_{n-1\ge j>i\ge 1}(a_j-a_i)$，即并不是左上角↖主对角的$D_{n-1}$，而是右下角↘主对角的$D_{n-1}$】

结论的记忆方法：可以用第二行来记，第二行某个后面的(靠右的)元素减去其左边的某个元素，所有的这样的不同组合相乘。——同样，如果给的是其转置形式的行列式，则第二行右减左的排列组合之积，将变为第二列下减上的排列组合之积。【第二行至少两个数相同←→行列式值为 0】

例 2.6.1 范得蒙德行列式的应用 1：

$\begin{vmatrix} 1 & x & x^2 & \cdots & x^{n-1} \\ 1 & a_1 & a_1^2 & \cdots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \cdots & a_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_{n-1} & a_{n-1}^2 & \cdots & a_{n-1}^{n-1} \end{vmatrix}=0$，其中$a_1,a_2,\ldots$互不相同，则 $x=a_1$或a_2或…。——同样，左到右是挺简单的，但是将"$(x-a_1)\cdot(x-a_2)\cdot\ldots\cdot(x-a_{n-1})*C_{n-1}^2$个常数因子"这样的多项式写成行列式的形式，就有价值了：于是$(x-a_1)\cdot(x-a_2)\cdot\ldots\cdot(x-a_{n-1})=(-1)^{n-1}\cdot$

$$\begin{vmatrix} 1 & x & x^2 & \cdots & x^{n-1} \\ 1 & a_1 & a_1^2 & \cdots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \cdots & a_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_{n-1} & a_{n-1}^2 & \cdots & a_{n-1}^{n-1} \end{vmatrix} / \begin{vmatrix} 1 & a_1 & a_1^2 & \cdots & a_1^{n-2} \\ 1 & a_2 & a_2^2 & \cdots & a_2^{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & a_{n-1} & a_{n-1}^2 & \cdots & a_{n-1}^{n-2} \end{vmatrix}$$ 。该结果可以与**例 2.3** 的结果进行比较。

例 2.6.2 范得蒙德行列式的应用 2：

如何得到这个类似范得蒙德行列式的行列式$\begin{vmatrix} 1 & 1 & 1 \\ a_1 & a_2 & a_3 \\ a_1^3 & a_2^3 & a_3^3 \end{vmatrix}$的值？

或许你想到了可以补一行 $a_1^2 \quad a_2^2 \quad a_3^2$，但要让它是方的，还得补一列 $\begin{matrix} 1 \\ x \\ x^2 \\ x^3 \end{matrix}$ 呀：

$\begin{vmatrix} 1 & 1 & 1 & 1 \\ a_1 & a_2 & a_3 & x \\ a_1^2 & a_2^2 & a_3^2 & x^2 \\ a_1^3 & a_2^3 & a_3^3 & x^3 \end{vmatrix}$，这样就能用此处的公式将其展开$=(x-a_1)\cdot(x-a_2)\cdot(x-a_3)*C_{4-1}^2$个常数因子。

另外，同时也按定义将其按最后一列展开$=1\cdot A_{14}+x\cdot A_{24}+x^2\cdot A_{34}+x^3\cdot A_{44}$，两式相等。

对 x 各次方的同类项进行对比可得，$A_{14}=(-a_1)(-a_2)(-a_3)\cdot3$ 个常数因子【或直接在原式中用定义$A_{14}=-\begin{vmatrix} a_1 & a_2 & a_3 \\ a_1^2 & a_2^2 & a_3^2 \\ a_1^3 & a_2^3 & a_3^3 \end{vmatrix}$】。

同样，x的一次方项的对比结果：$A_{24}=(a_1\cdot a_2+a_1\cdot a_3+a_2\cdot a_3)\cdot(a_3-a_2)\cdot(a_3-a_1)\cdot(a_2-a_1)$，

因此$\begin{vmatrix} 1 & 1 & 1 \\ a_1^2 & a_2^2 & a_3^2 \\ a_1^3 & a_2^3 & a_3^3 \end{vmatrix}=M_{24}=(a_1\cdot a_2+a_1\cdot a_3+a_2\cdot a_3)\cdot(a_3-a_2)\cdot(a_3-a_1)\cdot(a_2-a_1)$。同样，$x^2$项的对

比结果：A_{34}=$(-a_1-a_2-a_3)\cdot(a_3-a_2)\cdot(a_3-a_1)\cdot(a_2-a_1)$，因此$\begin{vmatrix} 1 & 1 & 1 \\ a_1 & a_2 & a_3 \\ a_1^3 & a_2^3 & a_3^3 \end{vmatrix}=M_{34}=$

$(a_1+a_2+a_3)\cdot(a_3-a_2)\cdot(a_3-a_1)\cdot(a_2-a_1)$。

同样的道理，可以给出类似的 n 阶行列式的值：$\begin{vmatrix} 1 & 1 & 1 & \cdots & 1 \\ a_1 & a_2 & a_3 & \cdots & a_n \\ a_1^2 & a_2^2 & a_3^2 & \cdots & a_n^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1^{n-2} & a_2^{n-2} & a_3^{n-2} & \cdots & a_n^{n-2} \\ a_1^n & a_2^n & a_3^n & \cdots & a_n^n \end{vmatrix}=$

$(a_1+a_2+\ldots+a_n)\cdot\prod_{1\le i<j\le n}(a_j-a_i)$。另外，看上去少了倒数第三行的 n 阶行列式的值，也可以类似地得出。

1.5 Cramer 法则

定理：若线性方程组的系数行列式 D≠0，方程组有唯一解：$x_i=\frac{D_i}{D}$。

证明：该定理的内容暗示，它的证明需要分为两点：一是解的存在性；二是解的唯一性。

[存在性] 在系数行列式$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$的左方和上方，分别先后增加一列$\begin{matrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{matrix}$一行

$b_i\ \ a_{i1}\ \ a_{i2}\ \cdots\ a_{in}$，即$D_{n+1}=\begin{vmatrix} b_i & a_{i1} & a_{i2} & \cdots & a_{in} \\ b_1 & a_{11} & a_{12} & \cdots & a_{1n} \\ b_2 & a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_n & a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$=0。将该$D_{n+1}$按第一行展开，则

$D_{n+1}=b_i\cdot D+a_{i1}\cdot A_{12}+\ldots+a_{in}\cdot A_{1,n+1}=0$，又因$A_{12}=-M_{12}=-D_1$、$A_{13}=M_{13}=-D_2$【对$M_{13}$的第 1 列做如下变换：一次次地将其与相邻的右列互换，直到将第一列挪到

$\begin{vmatrix} & & & \cdots & \\ b_1 & a_{11} & * & \cdots & a_{1n} \\ b_2 & a_{21} & * & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_n & a_{n1} & * & \cdots & a_{nn} \end{vmatrix}$因上一步展开而刚缺损的那一列上去，共移动 1 次，因此符号改变

一次】、$A_{1j}=(-1)^{1+j}\cdot M_{1j}=(-1)^{1+j}\cdot[(-1)^{j-2}D_{j-1}]=-D_{j-1}$，$j=2,3,\dots,n+1$。

因此$D_{n+1}=b_i\cdot D+a_{i1}\cdot(-D_1)+\dots+a_{in}\cdot(-D_n)=0$。又因 $D\neq0$，因此$a_{i1}\cdot\frac{D_1}{D}+\dots+a_{in}\cdot\frac{D_n}{D}=b_i$。当 i 取遍 1~n 时，即证明了解的存在性，即方程组有解，且解以$x_i=\frac{D_i}{D}$这样的存在方式存在，是合理可行的。——即$x_i=\frac{D_i}{D}$是方程组的解。但并不确定“方程组的解就是$x_i=\frac{D_i}{D}$”。

[唯一性] 设$x_j=d_j$是方程组的一组解，即$\sum_{j=1}^{n}a_{ij}d_j=b_i(i=1,2,\dots,n)$，我们来看看$d_j$们应该长什么样：

$$d_1\cdot D=\begin{vmatrix}a_{11}d_1 & a_{12} & \cdots & a_{1n}\\ a_{21}d_1 & a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ a_{n1}d_1 & a_{n2} & \cdots & a_{nn}\end{vmatrix}\xrightarrow[(j=2,3,\dots,n)]{c_1+d_j\cdot c_j}\begin{vmatrix}\sum_{j=1}^{n}a_{1j}d_j & a_{12} & \cdots & a_{1n}\\ \sum_{j=1}^{n}a_{2j}d_j & a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ \sum_{j=1}^{n}a_{nj}d_j & a_{n2} & \cdots & a_{nn}\end{vmatrix}=\begin{vmatrix}b_1 & a_{12} & \cdots & a_{1n}\\ b_2 & a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ b_n & a_{n2} & \cdots & a_{nn}\end{vmatrix}=D_1$$

。即解之一的d_1在数学表达式上必须满足：$d_1=\frac{D_1}{D}$；同理其他 j 下的$d_j=\frac{D_j}{D}$。

[额外部分]

1.齐次线性方程组：常数项全为 0 的线性方程组。

2.齐次线性方程组永远有解，零解就是其中一个(可用 Cramer 法则验证)。

3.推论：当齐次线性方程组的系数行列式不为 0 时，方程组只有零解。(齐次线性，说明有零解；D≠0，说明解唯一)

第四章将证明，当齐次线性方程组的系数行列式=0 时，方程组一定有非零解。

4.在用 Cramer 法则求解 n 元线性方程组时，需计算 n+1 个 n 阶行列式的值，计算量很大。而且它只适用于系数能构成行列式(比如 4 个 3 元一次方程，或是 3 个 4 元一次方程，所构成的方程组，其系数就看上去构不成行列式)，且系数行列式又≠零的那些线性方程组。它无法求解更一般的线性方程组。

1.6 行列式的应用

1.(1).$\begin{vmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$=0 是 x,y 的二元一次方程，表示一条平面直线。而$\begin{vmatrix} x_1 & y_1 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$=0、

$\begin{vmatrix} x_2 & y_2 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$=0，表示直线过点$(x_1,y_1)$、$(x_2,y_2)$。

因此$\begin{vmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$=0，表示平面上过点$(x_1,y_1)$、$(x_2,y_2)$的直线方程。

(2).于是，平面上三点共线的充要条件为：$\begin{vmatrix} x_3 & y_3 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$=0。

2.(1).同样的道理，三点所确定的平面方程为$\begin{vmatrix} x & y & z & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix}$=0。

(2).四点共面的充要条件：$\begin{vmatrix} x_4 & y_4 & z_4 & 1 \\ x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{vmatrix}$=0。

3.(1).函数插值问题：各学科领域广泛使用函数来表示变量间的数量关系，实际往往通过实验、检测等方法，获得函数在一些点上的函数值，而难以得到函数的解析表达式。那么，如何通过这些已知数据， 来得到函数的近似解析表达式，就很有实际意义了。

寻求近似函数的常用方法之一便是，插值方法。而插值方法最常用的是多项式插值。

插值方法：设函数 y=f(x)在区间[a,b]上连续，给定 n+1 个点$(a\le x_0<\ldots<x_n\le b)$上的函数值 $f(x_i)=y_i(i=0,1,\ldots n)$。要寻找一个 n 次多项式函数$p_n(x)$，作为 f(x)的近似函数，使得$p_n(x_i)=f(x_i)=y_i(i=0,1,\ldots n)$。这种寻求多项式$p_n(x)$的方法称为多项式插值方法。

(2).$p_n(x)=a_0+a_1x+\cdots+a_nx^n$，由 n+1 个系数唯一确定(这也是为什么我们要 n+1 个不同 x 值)。利用$p_n(x_i)=y_i(i=0,1,\ldots n)$，创造一个由 n+1 个方程构成的线性方程组。

其中，未知数是a_i们，已知的为 x 的各次方项们。系数行列式 D=$\begin{vmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{vmatrix}$=

$\prod_{0\le i<j\le n}(x_j - x_i)$。只要对于 i≠j 的$x_i \neq x_j$，即数据点(节点)互异时，就能通过 n+1 个数据点，利用 cramer 法则确定一个次数不超过 n 的插值多项式。

这个东西也叫作多项式拟合，可通过 n 次多项式以相关系数=1 地穿过 n+1 个目标点。在我的弹弹堂手游 B 型辅助里面拟合“拉杆滑移距离与炮弹初速度之间的非线性关系”时用到过一次。它实质上就是代入 n+1 个点的坐标，创造 n+1 个方程，解出 n 次多项式的 n+1 个待定系数。接下来的矩阵这块我也用到了二维旋转操作——“炮弹出生点相对于人物中心点以坡度为角度旋转”，这也被纳入了三种型号的弹弹堂辅助里。

Chapter02.矩阵 Matrix