第一个非零行起, 用第一类初等行变换可以把每个主元所在的列的其余元素变成零, 从而得到简化行阶梯矩阵. □

同理可以证明:

推论 1.4 任意一个矩阵都可以经过一系列初等列变换化成列阶梯矩阵或简化列阶梯矩阵.

历史寻根 在我国现存的最古老的数学书《九章算术》(成书于公元 1 世纪, 作者不详) 中, 有一章名为“方程”. ‘方’就是方形, ‘程’就是表达式. 将某个问题中的有关数据排成方形, 称为方程. 实际上就是现在的增广矩阵. 方程章的第一个问题是:

今有上禾 (稻棵) 三秉 (束), 中禾二秉, 下禾一秉, 实 (谷子) 三十九斗; 上禾二秉, 中禾三秉, 下禾一秉, 实三十四斗; 上禾一秉, 中禾二秉, 下禾三秉, 实二十六斗. 问上、中、下禾实一秉各几何.

翻译成现代的语言, 就是一个线性方程组:

$$\begin{cases} 3x+2y+\ \ z=39 \\ 2x+3y+\ \ z=34 \\ \ \ x+2y+3z=26 \end{cases}$$

在《九章算术》中使用了加减消元法解这个线性方程组, 并把系数排成如下的方形:

		左行	中行	右行
头位	上禾	1	2	3
中位	中禾	2	3	2
下位	下禾	3	1	1
	实	26	34	39

只要把上表横过来, 就是我们现在使用的增广矩阵形式.

上机实验 在 Maple 的线性代数软件包 linalg 中有以下一些与消元法有关的函数:

```
genmatrix(方程组, 变量组, flag);
augment(A, B, ...);
pivot(A, i, j);
```

```
gausselim(A);
rref(A);
backsub(A);
linsolve(A,B);
```

其中函数 genmatrix 从线性方程组生成系数矩阵或增广矩阵, 可选变量 flag 如果不出现, 就生成系数矩阵, 如果含有 flag, 就生成增广矩阵; 函数 augment 把矩阵 $A, B, \ldots$ 的列合在一起生成一个新矩阵; 函数 pivot 以矩阵 A 的 (i, j) 元作为主元, 把 A 的第 j 列的其余元消为 0; 函数 gausselim 把矩阵 A 化成行阶梯矩阵; 函数 rref 把矩阵 A 化成简化行阶梯矩阵; 函数 backsub 执行回代任务, 要求输入的 A 必须是行阶梯矩阵, 然后以 A 作为增广矩阵求出线性方程组的解向量; 函数 linsolve 求以 A 作为系数矩阵, B 作为常数项矩阵的线性方程组的解. 下列语句以例 1.1 的线性方程组为例, 演示了这些函数的用法.

```
>with(linalg):
>A:=matrix(4,4,[1,-1,-2,-5,-2,7,6,-12,3,-2,-5,-17,-5,-2,9,27]);
>B:=[10,6,31,-63];
>linsolve(A,B);
>C:=augment(A,B);
>pivot(C,1,1);
>pivot(",3,2);
>pivot(",2,3);
>F:=gausselim(C);
>backsub(F);
>G:=rref(C);
>backsub(G);
>genmatrix([x1-x2-2*x3-5*x4=10,-2*x1+7*x2+6*x3-12*x4=6,3*x1-2*x2
-5*x3-17*x4=31,-5*x1-2*x2+9*x3+27*x4=-63],[x1,x2,x3,x4],flag);
>rref(");
```

WIMS **网上游戏** 请读者到 WIMS 去试一试对谈式练习"直观高斯消元", 它将帮助你轻松地使用消元法解出线性方程组. 此外, 计算与作图工具中的"解线性方程组"能直接得出方程组的解.