一.计算方法及结果

求如下方程组的解:

$$Ax=\begin{pmatrix}-15\\27\\-23\\0\\-20\\12\\-7\\7\\10\end{pmatrix}\quad A=\begin{pmatrix}31&-13&0&0&0&-10&0&0&0\\-13&35&-9&0&-11&0&0&0&0\\0&-9&31&-10&0&0&0&0&0\\0&0&-10&79&-30&0&0&0&-9\\0&0&0&-30&57&-7&0&-5&0\\0&0&0&0&-7&47&-30&0&0\\0&0&0&0&0&-30&41&0&0\\0&0&0&0&-5&0&0&27&-2\\0&0&0&-9&0&0&0&-2&29\end{pmatrix}$$

1.Jacobi 迭代

迭代次数:33

数值解: -0.289235 0.34544 -0.712812 -0.220607 -0.430401 0.154313 -0.0578243 0.201054 0.290228

2.Gauss-Seidel 迭代，

迭代次数:15

数值解: -0.289227 0.345441 -0.712809 -0.220606 -0.430398 0.154313 -0.05782 0.201054 0.290229

二.算法分析

1. Jacobi 迭代计算公式

$$\begin{cases} x_1^{(k+1)} = -\frac{1}{a_{11}}(a_{12}x_2^{(k)} + \cdots + a_{1n}x_n^{(k)} - b_1) \\ x_2^{(k+1)} = -\frac{1}{a_{22}}(a_{21}x_1^{(k)} + a_{23}x_3^{(k)} + \cdots + a_{2n}x_n^{(k)} - b_2) \\ \vdots \\ x_n^{(k+1)} = -\frac{1}{a_{nn}}(a_{n1}x_1^{(k)} + \cdots + a_{n,n-1}x_{n-1}^{(k)} - b_n) \end{cases}$$

2. Guass-Seidel 迭代计算公式

$$\begin{cases} x_1^{(k+1)} = -\frac{1}{a_{11}}(a_{12}x_2^{(k)} + \cdots + a_{1n}x_n^{(k)} - b_1) \\ x_2^{(k+1)} = -\frac{1}{a_{22}}(a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)} + \cdots + a_{2n}x_n^{(k)} - b_2) \\ x_3^{(k+1)} = -\frac{1}{a_{33}}(a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)} + a_{34}x_4^{(k)} + \cdots + a_{3,n}x_n^{(k)} - b_3) \\ \vdots \\ x_n^{(k+1)} = -\frac{1}{a_{nn}}(a_{n1}x_1^{(k+1)} + a_{n2}x_2^{(k+1)} \cdots + a_{n,n-1}x_{n-1}^{(k+1)} - b_n) \end{cases}$$

3. 两种算法初始迭代 x(0)=(0,0,...,0)，停止条件|| x(k+1)-x(k) ||<=1e-5

4. 编程实现的说明
   使用 c 语言编程实现。其中 Jacobi()函数实现 Jocobi 迭代，Guass()函数实现 Guass-Seisel 迭代。在 main 函数中分别调用即可。

三.结果分析

1.从实验给定的方程组来看，Guass-Seidel 迭代法收敛速度较快。但是这个结论在一定条件下才是对的，甚至可能对于一些数据，Jocobi 算法迭代收敛而 Guass-Seidel 算法发散。
2. 两种算法都属于迭代求解方程组的近似解方法，都存在收敛性与精度控制问题.
3.两种方法计算所得结果基本相同，可以相互印证正确性

四.小结

1.计算机无法使用高斯消元法这样的纯数学方法来进行线性方程组的求解。计算机在解决此类问题时，采用迭代法。迭代法从解的某个近似值出发，通过构造一个无穷序列去逼近精确解。

2.实际上不是每个方程组都能迭代得到解，我们可以将系数矩阵转换为对角占优矩阵。