解线性方程组的迭代法

代码见lab5.cpp

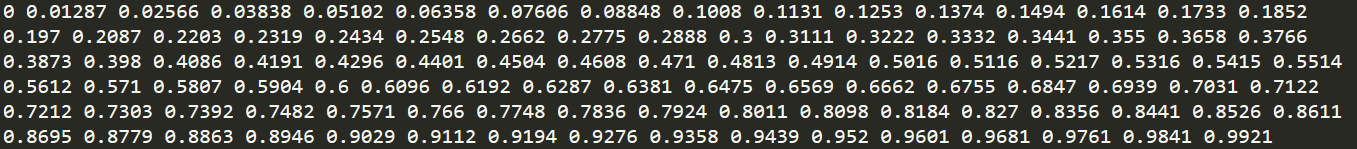
**代码说明**：

jacobi(),gauss\_seidel(),SOR()三个函数分别实现了三种不同的解线性方程组的迭代解法。在主函数中完成了线性方程组的构建与求解。其中jacobi与gauss\_seidel法同时实现了迭代版和递归版，递归版在注释中给出。

三个函数的参数意义相同，double\* jacobi(double \*a, double \*b, int n, int t)，a为系数矩阵，b为方程组的非线性部分，n为方程组的个数，即n x n 矩阵中的维数n，t为迭代次数。

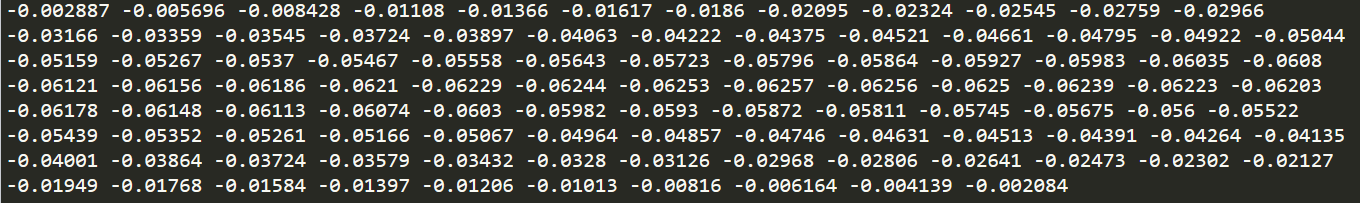
**运行结果：**

首先，方程的准确解应为：

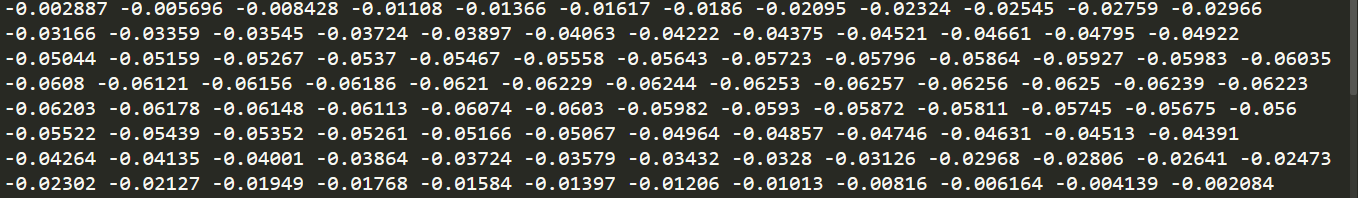


采用三种方法均迭代10000次的结果如下：

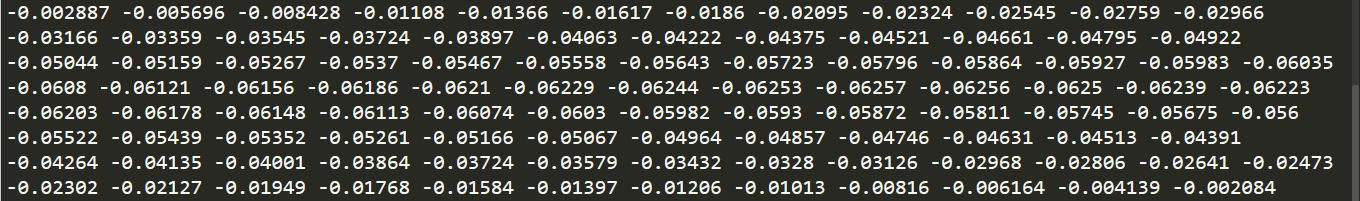
Jacobi：



Gauss\_seidel:

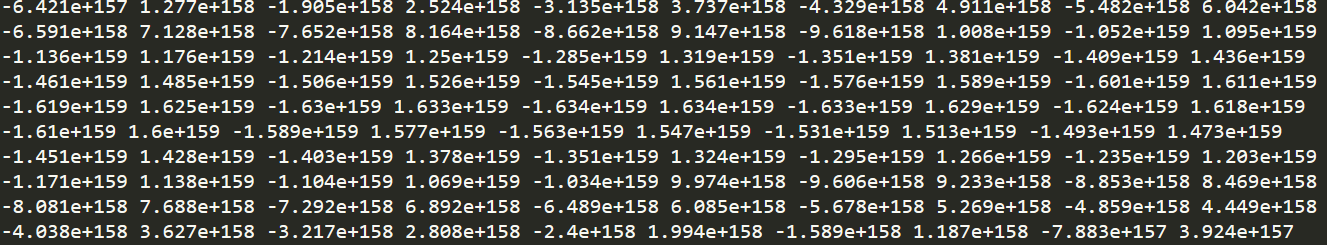


SOR(w = 0.5):



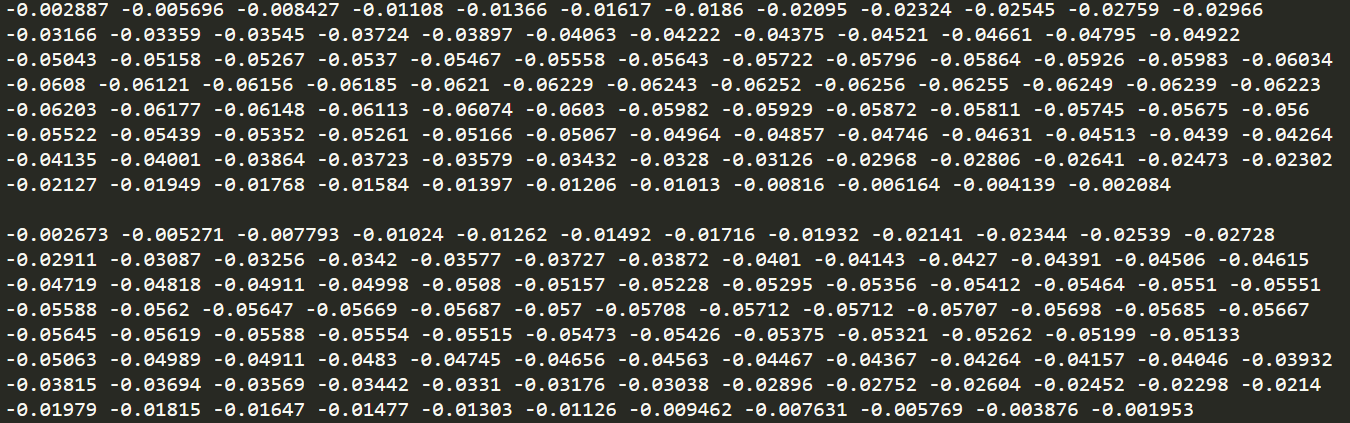
可以看出三种方法的结果都一样，但是都和准确值存在较大的误差，这种误差的主要来源推测应该是构建方程组时对导数的近似所导致的。

对于SOR法，将w改为1.3后，程序运行时间很久，且结果中出现nan，将w改为较接近1，如1.02，输出结果如下：



可见输出十分接近于0，继续增大w就将出现nan，且在收敛速度上也有很大变化。

减少迭代次数至1000次，jacobi法和gauss\_seidel法的输出结果和100000次迭代时相同，但是SOR法的结果出现了偏差，可见SOR法的收敛速度较慢。



继续减少迭代次数，jacobi法和gauss\_seidel法的输出始终都接近，可见两者的收敛速度相当。

**结论：**

从运行结果可以看出，jacobi法的gauss\_seidel法的收敛速度相当，且较SOR法更快。SOR法对w的取值很敏感，在本问题中，w大于1时，求解迅速趋向0从而使得迭代难以继续，结果中出现nan，且收敛速度出现很大的变化，从而看出SOR法的收敛速度对于w取值的依赖。

具体就本问题而言，采用迭代法求出的方程的解与准确值存在较大差异，考虑到计算的方法，应该把原因归结于构建方程组时对于其中导数的近似。从三种方法的求解结果较为一致，可以验证迭代法的正确性。