课程名称： 数值线性代数 **指导教师：** 刘兰冬

小组： 第 10 组 **姓名：** ________ **签字：** ________

实验项目名称：追赶法

实验目的及要求：

了解使用追赶法求解三对角方程组的原理和相关步骤，根据算法编写程序去求解一个具体的三对角方程组的解，再与之前的求解方法如列主元 Gauss 消去法比较二者求解结果的相关性质。

实验原理：

给定方程组 $A\vec{x}=\vec{f}$，其中 $A=\begin{pmatrix} b_1 & c_1 & & & \\ a_1 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{pmatrix}$，$\vec{x}=\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$，$\vec{f}=\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}$.

对矩阵 A 作 LU 分解，使 $A=LU$，其中 $L=\begin{pmatrix} 1 & & & & \\ \alpha_2 & 1 & & & \\ & \alpha_3 & \ddots & & \\ & & \ddots & 1 & \\ & & & \alpha_n & 1 \end{pmatrix}$，

$U=\begin{pmatrix} \beta_1 & \gamma_1 & & & \\ & \beta_2 & \gamma_2 & & \\ & & \ddots & \ddots & \\ & & & \beta_{n-1} & \gamma_{n-1} \\ & & & & \beta_n \end{pmatrix}$，比较两边元素得：$\begin{cases} b1=\beta_1 \\ a_2=\alpha_2\beta_1 \\ b_2=\alpha_2 c_1+\beta_2 \\ a_3=\alpha_3\beta_2 \\ b_3=\alpha_3 c_2+\beta_3 \end{cases}$，其中 $\gamma_i=c_i$，则能推出

$$\begin{cases}\beta_1 = b_1 \\ \alpha_2 = \dfrac{a_2}{\beta_1} \\ \beta_2 = b_2 - \alpha_2 c_1 \\ \alpha_3 = \dfrac{a_3}{\beta_2} \\ \beta_3 = b_3 - \alpha_3 c_2\end{cases}$$ 。最后总结公式为：$\begin{cases}\beta_1 = b_1 \\ \alpha_i = \dfrac{a_i}{\beta_{i-1}} \\ \beta_i = b_i - \alpha_i c_{i-1}\end{cases}$（$i = 2,3,...n$）。确定了矩阵 L，U 的元素之后，开始解下三角方程组 $L\vec{y} = \vec{f}$，再解上三角方程组 $U\vec{x} = \vec{y}$，即可解出目标方程组的解 $\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$。

实验内容（方法和步骤）：

利用列主元 Gauss 消去法和追赶法求解下列方程组，右端项自己构造。对 n=3, 10, 100, 1000（到你们所能求解的极限）进行求解，比较两种方法的存储量、计算时间和精度，，谈谈你们小组的感想。

$$\begin{pmatrix} -5 & 1 & & & \\ 1 & -5 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -5 & 1 \\ & & & 1 & -5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}$$

① *使用列主元 Gauss 消去法求解*

```
function [n,Time,wucha]=lzy(n)
digits(6);
tic;
[A,z,b]=matrix(n);
detA = 1;
q=0;
for k=1:1:n-1
    s=0;
    for i=k:1:n
        if(abs(A(i,k))>s)
            s = abs(A(i,k));
            q=i;
        end
```

```
    end
    if(A(q,k)==0)
          return ;
    end
    if(q~=k)
        for j=k:1:n
            temp = A(k,j);
            A(k,j) = A(q,j);
            A(q,j) = temp;
        end
        temp = b(k);
        b(k)   = b(q);
        b(q)   =  temp;
        detA = -1*detA;
    end
     for i=k+1:1:n
                A(i,k)=A(i,k)/A(k,k);
                for j=k+1:1:n
                      A(i,j)=A(i,j)-A(i,k)*A(k,j);
                end
              b(i)=b(i)-A(i,k)*b(k);
     end
  detA = detA*A(k,k);
end
  if(A(n,n)==0)
     return ;
  else
     detA  =detA *A(n,n);
  end
 x(n)=b(n)/A(n,n);
   for i=n-1:-1:1
      sum=0;
      for j=i+1:1:n
         sum=sum+A(i,j)*x(j);
      end
      x(i)=(b(i)-sum)/A(i,i);
   end
   wucha=norm(x-z);
Time=toc;%计时结束
Time=vpa(Time);
```

② *使用追赶法求解*

```
function [n,Time,wucha]=zg(n)%输入想求的阶数，输出，阶数，时间，误差（默认小数点后六位）
```

```
digits(6);%括号里的数字就是小数点后保留几位
tic;%计时开始
[A,z,f]=matrix(n);%调用函数 matrix
[a,b,c]=transform(A,n);%调用函数 transform
%step 1 追
beta(1)=b(1);
for i=2:n
    alpha(i)=a(i)/beta(i-1);
    beta(i)=b(i)-alpha(i)*c(i-1);
end
%step 2 追
y=0;
y(1)=f(1);
for i=2:n
    y(i)=f(i)-alpha(i)*y(i-1);
end
%step 3 赶
x=0;
x(n)=y(n)/beta(n);
for i=n-1:-1:1
    x(i)=(y(i+1)-c(i)*x(i+1))/beta(i);
end
wucha=norm(x-z)
Time=toc%计时结束
Time=vpa(Time)
```

※ 以上 2 个程序均会调用以下 2 个打包好的程序：

①

```
function [A,z,f]=matrix(n)
for i=1:n
    z(i)=1;
end
for i=1:n
    A(i,i)=-5;
end
for i=2:n
    A(i,i-1)=1;
end
for i=2:n-1
    A(i-1,i)=1;
end
f(1)=A(1,1)*z(1)+A(1,2)*z(2);
for i=2:n-1
    f(i)=A(i,i-1)*z(i-1)+A(i,i)*z(i)+A(i,i+1)*z(i+1);
```

```
end
f(n)=A(n,n-1)*z(n-1)+A(n,n)*z(n);
```

②

```
function [a,b,c]=transform(A,n)
b(1)=A(1,1);
c(1)=A(2,1);
for i=2:n-1
    a(i)=A(i,i-1);
    b(i)=A(i,i);
    c(i)=A(i,i+1);
end
a(n)=A(n,n-1);
b(n)=A(n,n);
```

附：一维存储方式的追赶法程序

```
function [Time,wucha,n]=jqzg(n)
digits(6);
tic;
for i=1:n
    a(i)=1;
    b(i)=-5;
    c(i)=1;
    r(i)=1;
end
f(1)=b(1)*r(1)+c(1)*r(2);
for i=2:n-1
    f(i)=a(i)*r(i-1)+b(i)*r(i)+c(i)*r(i+1);
end
f(n)=a(n)*r(n-1)+b(n)*r(n);
s(1)=b(1);
for i=2:n
    t(i)=a(i)/s(i-1);
    s(i)=b(i)-t(i)*c(i-1);
end
y(1)=f(1);
for i=2:n
    y(i)=f(i)-t(i)*y(i-1);
end
x(n)=y(n)/s(n);
for i=n-1:-1:1
    x(i)=(y(i)-c(i)*x(i+1))/s(i);
end
Time=toc;
wucha=norm(x-r);
```

```
Time=vpa(Time);
```

实验结果与分析：

依次增大 n 的阶数，运行得出 2 种方法解方程组所用时间及计算精度，并整理成如下表格:

表 1

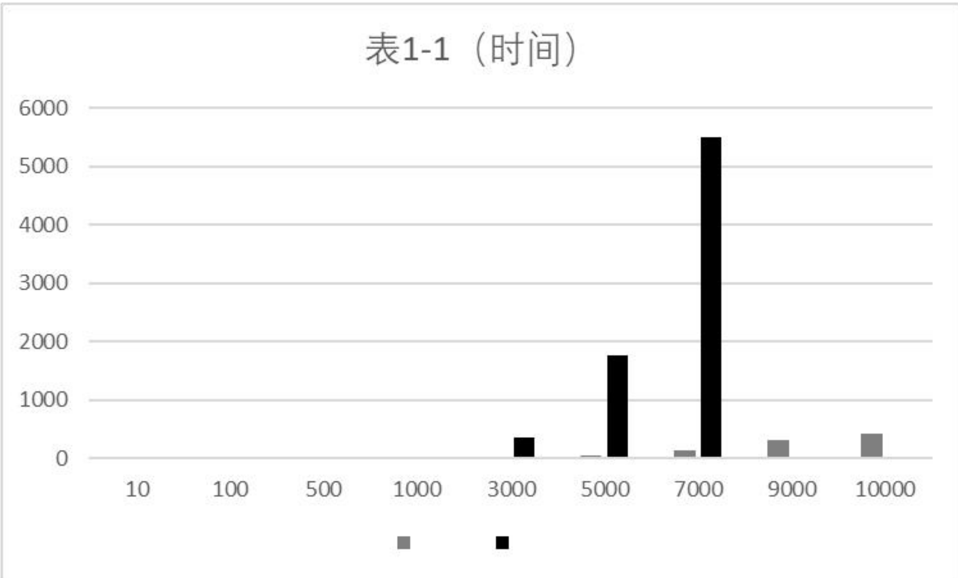
n	计算时间（追赶法）	计算时间（列主元）	精度（追赶法）	精度（列主元）
10	0.000922075	0.0000920495	3.8459e-16	3.8459e-16
100	0.00112237	0.0071925	3.8459e-16	2.1065e-15
500	0.0230779	1.0617	4.9202e-15	4.9152e-15
1000	0.178419	9.40801	6.9900e-15	6.9865e-15
3000	10.2816	357.422	1.2144e-14	1.2142e-14
5000	49.6849	1762.15	1.5687e-14	1.5685e-14
7000	141.489	5500.84	1.8566e-14	1.8564e-14
9000	303.893		2.1054e-14	
10000	416.328		2.2194e-14	

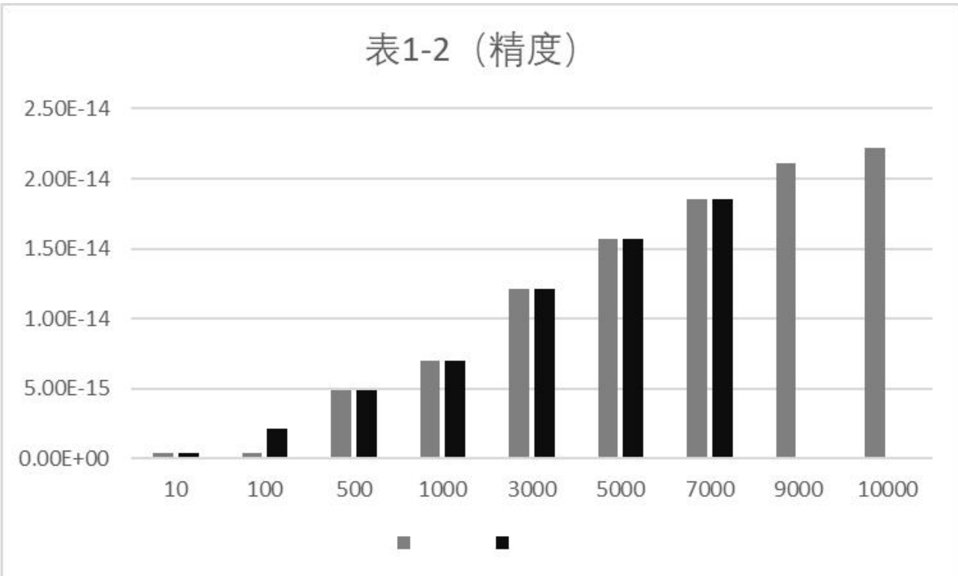
两种方法的各项比较:

表 2

<table>
<tr><th>方法</th><th>存储量</th><th colspan="2">计算时间(s)</th><th colspan="2">误差</th></tr>
<tr><td rowspan="5">列主元 Gauss 消去法（二维）</td><td rowspan="5">n^2+2n</td><td>3 阶</td><td>0.0322</td><td>3 阶</td><td>0</td></tr>
<tr><td>10 阶</td><td>0.0788831</td><td>10 阶</td><td>3.8459e-16</td></tr>
<tr><td>100 阶</td><td>0.0090714</td><td>100 阶</td><td>2.1065e-15</td></tr>
<tr><td>1000 阶</td><td>9.3313</td><td>1000 阶</td><td>6.9865e-15</td></tr>
<tr><td>10000 阶</td><td></td><td>10000 阶</td><td></td></tr>
<tr><td rowspan="2">追赶法（一维）</td><td rowspan="2">$5n-2$</td><td>3 阶</td><td>0.0000940246</td><td>3 阶</td><td>0</td></tr>
<tr><td>10 阶</td><td>0.0267607</td><td>10 阶</td><td>3.8459e-16</td></tr>
</table>

		100 阶	0.0157321	100 阶	2.1182e-15
		1000 阶	0.0230795	1000 阶	6.9900e-15
		10^8 阶	96.9558	10^8 阶	2.2204e-12
追赶法（二维）	n^2+2n	3 阶	0.0000940246	3 阶	0
		10 阶	0.000922075	10 阶	3.8459e-16
		100 阶	0.00112237	100 阶	3.8459e-16
		1000 阶	0.178419	1000 阶	6.9900e-15
		10^8 阶		10^8 阶	





实验感想：

对于本次实验来讲，由于解方程的算法并无区别，所以精度方面基本无差别。在相同存储结构下来讲，追赶法比列主元法运行速度更快。对于追赶法，不同的存储结构导致运行时间差别在高阶情况下很明显。总体来说，在相同存储结构下，追赶法比列主元法要快，精度不相上下。

成绩：　　　　　　　　批阅教师签名：　　　　　　　　2017 年　10　月　17　日