实验一：通过乱数产生3\*3的Ax阵列，以及1\*3的b阵列

|  |
| --- |
| -3.29112 8.77041 -3.23496 -9.31181  -8.22474 9.83856 -5.0499 7.69036  -4.8558 7.31315 -4.72243 -2.85073 |

经过流程图1 将其转换为倒三角阵列

|  |
| --- |
| -3.29112 8.77041 -3.23496 -9.31181  0 -12.0793 3.03451 30.9612  0 0 -1.36306 -3.53459 |

最后通过流程图2解出x1, x2, x3

|  |
| --- |
| -4.81402 -1.91172 2.59314 |

实验二

：通过乱数产生5\*5的Ax阵列，以及1\*5的b阵列

|  |
| --- |
| -9.63713 4.96231 -3.95825 8.20185 -8.33369 -0.0320444  -4.67879 4.44105 -6.70431 6.43117 -9.19919 9.24131  -5.55437 6.00146 -0.646687 0.17304 -1.94189 -4.43892  -8.37916 0.473952 -5.06424 8.38954 -2.96182 2.67098  -3.0958 3.65581 -6.545 6.80166 -6.36433 -4.59944 |

经过流程图1 将其转换为倒三角阵列

|  |
| --- |
| -9.63713 4.96231 -3.95825 8.20185 -8.33369 -0.0320444  0 2.03187 -4.78259 2.4492 -5.15322 9.25687  0 0 9.02892 -8.34076 10.8285 -18.7323  0 0 0 -3.96222 7.33139 -1.92579  0 0 0 0 4.43898 -15.4832 |

最后通过流程图2解出x1~x5

|  |
| --- |
| -3.29255 -5.1103 -3.40452 -5.96791 -3.48801 |

实验三：通过乱数产生10\*10的Ax阵列，以及1\*10的b阵列

|  |
| --- |
| -0.686056 8.60805 -1.14139 1.78777 -8.67733 5.81591 -3.10373 3.75561 -3.36741 7.48161 -0.340892  -8.33705 5.04044 -5.53423 2.00171 -4.46883 9.30631 -9.54039 5.21867 -2.18085 4.88296 6.08875  -4.26801 3.86792 -6.11438 8.30226 -3.7608 9.90692 -3.13669 5.74358 -4.13312 6.03259 -4.46974  -7.03574 5.20432 -8.31233 7.70592 -3.70128 7.20328 -5.17624 1.98828 -5.85833 6.83309 9.16868  -4.64766 8.51253 -1.41148 4.97177 -5.30717 2.88949 -4.46944 1.46886 -6.08783 7.87408 -0.40437  -3.10221 8.11457 -8.61782 2.59194 -2.90323 4.20667 -1.76458 5.15213 -0.413831 2.52937 1.12308  -4.95376 8.94253 -5.47227 8.55434 -1.81677 7.93756 -6.55171 0.673238 -0.73397 1.31626 -9.41404  -8.14997 4.28571 -3.83587 2.71371 -4.29121 7.53838 -1.02145 4.40901 -7.2042 1.54302 7.02506  -0.561235 1.77129 -5.23789 2.87851 -2.8428 8.27174 -9.07987 1.18686 -8.82504 6.37165 -0.272225  -0.334483 9.68627 -9.75951 4.60402 -1.53172 1.39836 -9.74761 6.16077 -1.6187 8.91934 6.69881 |

经过流程图1 将其转换为倒三角阵列

|  |
| --- |
| -0.686056 8.60805 -1.14139 1.78777 -8.67733 5.81591 -3.10373 3.75561 -3.36741 7.48161 -0.340892  0 -99.5658 8.33613 -19.7236 100.979 -61.3694 28.1766 -40.42 38.7404 -86.0346 10.2313  0 0 -3.17343 7.02246 -0.167229 4.3491 2.11172 2.54928 -2.5157 2.42027 -7.45447  -4.76837e-007 0 -2.38419e-007 -2.05472 1.22186 -6.11845 0.773516 -5.66343 -0.824007 -0.826018 12.496  0 -3.8147e-006 0 0 7.30621 -25.1582 6.71266 -22.6623 -7.36081 -1.14468 37.2639  0 0 0 0 0 19.0937 -3.26638 24.2164 10.5313 -4.49887 -52.7498  0 0 1.19209e-007 0 0 0 -4.03468 -9.81944 1.72101 -3.41103 13.5672  0 0 0 0 0 0 0 -18.1506 0.0285161 -11.0319 26.4003  0 0 0 0 0 -4.76837e-007 0 0 -12.9553 2.49381 -2.01152  0 0 0 0 0 0 0 0 0 -2.7345 16.6208 |

最后通过流程图2解出下x1~x10

|  |
| --- |
| -0.213957 -4.94595 -2.91732 3.95835 -10.8708 -7.17591 -4.10402 2.23819 -1.01474 -6.07819 |

以下为实现该功能的代码

#include <stdio.h>

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <math.h>

using namespace std;

int main(){

int n;

cin>>n;

srand((unsigned)time(NULL));

float a[n][n+1],x[n],sum,d,c=1;

for(int i=0;i<n;i++){

c=-1\*c;

a[i][n] = 10\*c\*rand() / double(RAND\_MAX);

} //B的值设好

for(int i=0;i<n;i++){

c=1;

for(int j=0;j<n;j++){

c=-1\*c;

a[i][j] = 10\*c\*rand() / double(RAND\_MAX);

}

} //A阵列的值设好

for(int i=0;i<n;i++) {

for(int j=0;j<n+1;j++){

cout<<a[i][j]<<" ";

}

cout<<endl;

}

for(int k=0;k<n-1;k++){

for(int i=k+1;i<n;i++){

d=a[i][k]/a[k][k];

for(int j=k;j<n+1;j++){

a[i][j] = a[i][j] - d\*a[k][j];

}

}

} //变成倒三角

for(int i=0;i<n;i++) {

for(int j=0;j<n+1;j++){

cout<<a[i][j]<<" ";

}

cout<<endl;

}

for(int i=n-1;i>=0;i--){

sum=0;

for(int j=i+1;j<n;j++)

sum= sum+ a[i][j]\*x[j];

x[i] = (a[i][n]-sum) / a[i][i];

} //求解

for(int i=0;i<n;i++){

cout<<x[i]<<" ";

}

cout<<endl;

return 0;

}