数值分析方法

作业3

Problem 1

解: a. $\because \mathbf{x} = (0, -7, 5)^t$, $\tilde{\mathbf{x}} = (-0.2, -7.5, 5.4)^t$

$\therefore \mathbf{x} - \tilde{\mathbf{x}} = (0.2, 0.5, -0.4)^t$

$\therefore ||\mathbf{x} - \tilde{\mathbf{x}}||_\infty = 0.5$

将$\tilde{\mathbf{x}}$代入方程组，可得:

$$A\tilde{\mathbf{x}} - \mathbf{b} = \begin{pmatrix} 0 \\ -0.3 \\ -0.2 \end{pmatrix}$$

$\therefore ||A\tilde{\mathbf{x}} - \mathbf{b}||_\infty = 0.3$

b. $\because \mathbf{x} = (0, -7, 5)^t$, $\tilde{\mathbf{x}} = (-0.33, -7.9, 5.8)^t$

$\therefore \mathbf{x} - \tilde{\mathbf{x}} = (0.33, 0.9, -0.8)^t$

$\therefore ||\mathbf{x} - \tilde{\mathbf{x}}||_\infty = 0.9$

将$\tilde{\mathbf{x}}$代入方程组，可得:

$$A\tilde{\mathbf{x}} - \mathbf{b} = \begin{pmatrix} 0.27 \\ -0.16 \\ 0.21 \end{pmatrix}$$

$\therefore ||A\tilde{\mathbf{x}} - \mathbf{b}||_\infty = 0.27$

Problem 2

解: $\because \mathbf{A}$是对称矩阵,则$\mathbf{A}^T = \mathbf{A}$

又 $||\mathbf{A}||_2 = \sqrt{\rho(\mathbf{A}\mathbf{A}^T)}$

$\therefore ||\mathbf{A}||_2 = \sqrt{\rho(\mathbf{A}^2)}$

由定义可知, $\rho(\mathbf{A}) = \max|\lambda| > 0$, 其中$\lambda$为$\mathbf{A}$的特征根

$\therefore \rho(\mathbf{A}^2) = \max|\lambda^2| = \big(\max|\lambda|\big)^2 = \rho^2(\mathbf{A})$

$\therefore ||\mathbf{A}||_2 = \sqrt{\rho^2(\mathbf{A})} = \rho(\mathbf{A})$

证毕.

Problem 3

解：代码如下:

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
        int n;
        scanf("%d",&n);
        double a[n][n+1];
        double s[n],m[n][n];
        double max,sum;
        double x[n];
        int i=0,j=0,k,p;
        int row[n],temp;
        while (i<n*(n+1)){
                scanf("%lf",&a[0][i]);
                i++;
        }
        for (i=0;i<n;i++){
                j=0;
                s[i]=fabs(a[i][0]);
                while (j<n){
                        if (fabs(a[i][j])>s[i]){
                                s[i]=fabs(a[i][j]);
                        }
                        if (s[i]==0){
                                printf("Algorithm \
 failed");
                                exit(0);
                        }
                j++;
                }
                row[i]=i;
        }
        for (i=0;i<n-1;i++){
                max=a[row[i]][i]/s[row[i]];
                for (j=i;j<n;j++){
                        if (max<a[row[j]][i]/\
                                s[row[j]]){
                                max=a[row[j]][i]/\
                                s[row[j]];
                                p=j;
```

```
			}
		}
		if (a[row[p]][i]==0){
			printf("Algorithm failed");
			exit(0);
		}
		if (row[i]!=row[p]){
			temp=row[i];
			row[i]=row[p];
			row[p]=temp;
		}
		for (j=i+1;j<n;j++){
			m[row[j]][i]=a[row[j]][i]/\
			a[row[i]][i];
			for (k=0;k<=n;k++){
				a[row[j]][k]-=m[row\
				[j]][i]*a[row[i]][k];
			}
		}
	}
	if (a[row[n-1]][n-1]==0){
		printf("Algorithm failed");
		exit(0);
	}
	x[n-1]=a[row[n-1]][n]/a[row[n-1]][n-1];
	for (i=n-2;i>=0;i--){
		sum=0;
		for (j=i+1;j<n;j++){
			sum+=x[j]*a[row[i]][j];
		}
		x[i]=(a[row[i]][n]-sum)/a[row[i]][i];
	}
	for (i=0;i<n;i++){
		printf("x[%d]=%.8lf\n",i+1,x[i]);
	}
}
```

a. 由题意：增广矩阵为

$$\tilde{\mathbf{A}} = \begin{pmatrix} 0.03 & 58.9 & 59.2 \\ 5.31 & -6.10 & 47.0 \end{pmatrix}$$

代入程序可得: $\tilde{\mathbf{x}} = (10.00000000, 1.00000000)^T$.

b. 由题意: 增广矩阵为

$$\tilde{\mathbf{A}} = \begin{pmatrix} 3.03 & -12.1 & 14 & -119 \\ -3.03 & 12.1 & -7 & 120 \\ 3.11 & -14.2 & 21 & -139 \end{pmatrix}$$

代入程序可得: $\tilde{\mathbf{x}} = (0.00000000, 10.00000000, 0.14285714)^T$.

Problem 4

解: 代码如下:

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
        int n;
        scanf("%d",&n);
        double a[n][n];
        double b[n];
        double x0[n],x[n];
        double sum=0;
        int i,j,k=0;
        for (i=0;i<n*n;i++){
                scanf("%lf",&a[0][i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&b[i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&x0[i]);
        }
        while (k<3){
                for (i=0;i<n;i++){
                        sum=0;
                        for (j=0;j<n;j++){
                                if (j!=i){
                                        sum+=a[i][j]*x0[j];
                                }
                        }
```

```
                        x[i]=(b[i]-sum)/a[i][i];
                }
                for (i=0;i<n;i++){
                        x0[i]=x[i];
                }
                k++;
                printf("n = %d, x = ( ",k-1);
                for (i=0;i<n-1;i++){
                        printf("%10.8lf,",x[i]);
                }
                printf("%10.8lf )^T\n",x[n-1]);
        }
}
```

a. 由题意:

$$\mathbf{A} = \begin{pmatrix} 4 & 1 & -1 \\ -1 & 3 & 1 \\ 2 & 2 & 5 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 5 \\ -4 \\ 1 \end{pmatrix}$$

$\because \mathbf{x}^{(0)} = \mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	1.25000000	-1.33333333	0.20000000
2	1.63333333	-0.98333333	0.23333333
3	1.55416667	-0.86666667	-0.06000000

故近似解为 $\tilde{\mathbf{x}} = (1.55416667, -0.86666667, -0.06000000)^T$.

b. 由题意:

$$\mathbf{A} = \begin{pmatrix} -2 & 1 & 0.5 \\ 1 & -2 & -0.5 \\ 0 & 1 & 2 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 4 \\ -4 \\ 0 \end{pmatrix}$$

$\because \mathbf{x}^{(0)} = \mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	-2.00000000	2.00000000	0.00000000
2	-1.00000000	1.00000000	-1.00000000
3	-1.75000000	1.75000000	-0.50000000

故近似解为 $\tilde{\mathbf{x}} = (-1.75000000, 1.75000000, -0.50000000)^T$.

Problem 5

解：当使用雅可比迭代时,代码如下:

```c
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
        int n;
        scanf("%d",&n);
        double a[n][n];
        double b[n];
        double x0[n],x[n];
        double sum=0;
        double norm[n],norml,tol;
        int i,j,k=1;
        for (i=0;i<n*n;i++){
                scanf("%lf",&a[0][i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&b[i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&x0[i]);
        }
        scanf("%lf",&tol);
        while (k){
                for (i=0;i<n;i++){
                        sum=0;
                        for (j=0;j<n;j++){
                                if (j!=i){
                                        sum+=a[i]\
                                         [j]*x0[j];
                                }
                        }
                        x[i]=(b[i]-sum)/a[i][i];
                        norm[i]=fabs(x[i]-x0[i]);
                }
                for (i=0;i<n;i++){
                        x0[i]=x[i];
                }
                k++;
```

```c
                printf("n = %d, x = ( ",k-1);
                for (i=0;i<n-1;i++){
                        printf("%10.8lf, ",x[i]);
                }
                printf("%10.8lf ), ",x[n-1]);
                //Output Ax-b
                printf("Ax-b = ( ");
                for (i=0;i<n-1;i++){
                        sum=0;
                        for (j=0;j<n;j++){
                                sum+=a[i][j]*x[j];
                        }
                        sum-=b[i];
                        printf("%10.8lf, ",sum);
                }
                sum=0;
                for (j=0;j<n;j++){
                        sum+=a[i][j]*x[j];
                }
                sum-=b[i];
                printf("%10.8lf )^T\n",sum);
                norml=norm[0];
                for (i=0;i<n;i++){
                        if (norm[i]>norml){
                                norml=norm[i];
                        }
                }
                if (norml<=tol){
                        break;
                }
        }
}
```

a. 由题意:

$$\mathbf{A}=\begin{pmatrix}3 & -1 & 1\\ 3 & 6 & 2\\ 3 & 3 & 7\end{pmatrix},\mathbf{b}=\begin{pmatrix}1\\0\\4\end{pmatrix}$$

令$\mathbf{x}^{(0)}=\mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	0.33333333	0.00000000	0.57142857
2	0.14285714	-0.35714286	0.42857143
3	0.07142857	-0.21428571	0.66326531
4	0.04081633	-0.25680272	0.63265306
5	0.03684807	-0.23129252	0.66399417
6	0.03490444	-0.23975543	0.65476190
7	0.03516089	-0.23570619	0.65922185
8	0.03502399	-0.23732106	0.65737656
9	0.03510079	-0.23663751	0.65812732

故近似解为 $\tilde{\mathbf{x}} = (0.03510079, -0.23663751, 0.65812732)^T$.

b. 由题意:

$$\mathbf{A} = \begin{pmatrix} 10 & -1 & 0 \\ -1 & 10 & -2 \\ 0 & -2 & 10 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 9 \\ 7 \\ 6 \end{pmatrix}$$

令$\mathbf{x}^{(0)} = \mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	0.90000000	0.70000000	0.60000000
2	0.97000000	0.91000000	0.74000000
3	0.99100000	0.94500000	0.78200000
4	0.99450000	0.95550000	0.78900000
5	0.99555000	0.95725000	0.79110000
6	0.99572500	0.95777500	0.79145000

故近似解为 $\tilde{\mathbf{x}} = (0.99572500, 0.95777500, 0.79145000)^T$.

当使用高斯-赛德尔迭代时,代码如下:

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
        int n;
        scanf("%d",&n);
        double a[n][n];
        double b[n];
        double x0[n],x[n];
        double sum1=0,sum2=0,sum;
        double norm[n],norml,tol;
```

```c
        int i,j,k=1;
        for (i=0;i<n*n;i++){
                scanf("%lf",&a[0][i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&b[i]);
        }
        for (i=0;i<n;i++){
                scanf("%lf",&x0[i]);
        }
        scanf("%lf",&tol);
        while (k){
                for (i=0;i<n;i++){
                        sum1=0;
                        sum2=0;
                        for (j=0;j<i;j++){
                                sum1+=a[i][j]*x[j];
                        }
                        for (j=i+1;j<n;j++){
                                sum2+=a[i][j]*x0[j];
                        }
                        x[i]=(b[i]-sum1-sum2)/a[i]\
                             [i];
                        norm[i]=fabs(x[i]-x0[i]);
                }
                for (i=0;i<n;i++){
                        x0[i]=x[i];
                }
                k++;
                printf("n = %d, x = ( ",k-1);
                for (i=0;i<n-1;i++){
                        printf("%10.8lf,",x[i]);
                }
                printf("%10.8lf ),",x[n-1]);
                //Output Ax-b
                printf("Ax-b = ( ");
                for (i=0;i<n-1;i++){
                        sum=0;
                        for (j=0;j<n;j++){
                                sum+=a[i][j]*x[j];
                        }
```

```
            sum-=b[i];
            printf("%10.8lf,",sum);
        }
        sum=0;
        for (j=0;j<n;j++){
            sum+=a[i][j]*x[j];
        }
        sum-=b[i];
        printf("%10.8lf )\n",sum);
        norml=norm[0];
        for (i=0;i<n;i++){
            if (norm[i]>norml){
                norml=norm[i];
            }
        }
        if (norml<=tol){
            break;
        }
    }
}
```

a. 由题意:

$$\mathbf{A}=\begin{pmatrix}3 & -1 & 1\\ 3 & 6 & 2\\ 3 & 3 & 7\end{pmatrix},\mathbf{b}=\begin{pmatrix}1\\ 0\\ 4\end{pmatrix}$$

令$\mathbf{x}^{(0)}=\mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	0.33333333	-0.16666667	0.50000000
2	0.11111111	-0.22222222	0.61904762
3	0.05291005	-0.23280423	0.64852608
4	0.03955656	-0.23595364	0.65559875
5	0.03614920	-0.23660752	0.65733928
6	0.03535107	-0.23678863	0.65775895

故近似解为 $\tilde{\mathbf{x}}=(0.03535107,-0.23678863,0.65775895)^T$.

b. 由题意:

$$\mathbf{A}=\begin{pmatrix}10 & -1 & 0\\ -1 & 10 & -2\\ 0 & -2 & 10\end{pmatrix},\mathbf{b}=\begin{pmatrix}9\\ 7\\ 6\end{pmatrix}$$

令$\mathbf{x}^{(0)}=\mathbf{0}$

可得:

n	$x_1^{(n)}$	$x_2^{(n)}$	$x_3^{(n)}$
1	0.90000000	0.79000000	0.75800000
2	0.97900000	0.94950000	0.78990000
3	0.99495000	0.95747500	0.79149500
4	0.99574750	0.95787375	0.79157475

故近似解为 $\tilde{\mathbf{x}} = (0.99574750, 0.95787375, 0.79157475)^T$.

Problem 6

解: 由数学归纳法:

当$n=1$时, $\because \boldsymbol{\chi}_1 \neq \mathbf{0}$

则 $\boldsymbol{\chi}_1$线性无关

假设$n=k-1$时, $\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_{k-1}$线性无关

当$n=k$时,设有

$$c_1\boldsymbol{\chi}_1 + c_2\boldsymbol{\chi}_2 + \cdots + c_{k-1}\boldsymbol{\chi}_{k-1} + c_k\boldsymbol{\chi}_k = \mathbf{0} \tag{1}$$

(1)式两边同乘ρ_k, 得:

$$c_1\rho_k\boldsymbol{\chi}_1 + c_2\rho_k\boldsymbol{\chi}_2 + \cdots + c_{k-1}\rho_k\boldsymbol{\chi}_{k-1} + c_k\rho_k\boldsymbol{\chi}_k = \mathbf{0} \tag{2}$$

$\because \mathbf{A}\boldsymbol{\chi}_i = \rho_i\boldsymbol{\chi}_i$

(1)式两边同乘$\mathbf{A}$, 则:

$$c_1\rho_1\boldsymbol{\chi}_1 + c_2\rho_2\boldsymbol{\chi}_2 + \cdots + c_{k-1}\rho_{k-1}\boldsymbol{\chi}_{k-1} + c_k\rho_k\boldsymbol{\chi}_k = \mathbf{0} \tag{3}$$

用(2)减去(3)式, 得:

$$c_1(\rho_k - \rho_1)\boldsymbol{\chi}_1 + c_2(\rho_k - \rho_2)\boldsymbol{\chi}_2 + \cdots + c_{k-1}(\rho_k - \rho_{k-1})\boldsymbol{\chi}_{k-1} = \mathbf{0} \tag{4}$$

$\because \boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_{k-1}$线性无关, 且特征根两两互异

$\therefore c_1 = c_2 = \cdots = c_{k-1} = 0$

代入(3)式得: $c_k = 0$

故$\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \ldots, \boldsymbol{\chi}_{k-1}, \boldsymbol{\chi}_k$线性无关, 证毕.

Problem 7

解: 假设严格对角占优矩阵$\mathbf{A}$不可逆, 则$\mathbf{A}\mathbf{x} = \mathbf{0}$存在无穷多个解

记$\mathbf{A}$中某一行为$\{a_{kj}\}, 1 \leq k \leq n,\ j = 1, 2, \ldots, n$

设其一解为$\mathbf{x} = \{x_1, x_2, \ldots, x_n\}^T$, 且满足$|x_k| = \max\{|x_1|, |x_2|, \ldots, |x_n|\}$, 则:

$$\sum_{j=1}^{n} a_{kj}x_j = 0$$

于是:

$$\sum_{j\neq k}^{n} a_{kj}x_j = -a_{kk}x_k$$

$$\therefore |a_{kk}x_k| = \left|\sum_{j\neq k}^{n} a_{kj}x_j\right|$$

又

$$|a_{kk}x_k| = |a_{kk}||x_k| > \left|\sum_{j\neq k}^{n} a_{kj}\right||x_k| \geq \sum_{j\neq k}^{n} |a_{kj}||x_j| \geq \left|\sum_{j\neq k}^{n} a_{kj}x_j\right|$$

矛盾, 假设不成立.

故严格对角占优矩阵必可逆.