

Numerische Mathematik

4. Beleg

1.) Lösung überbestimmter linearer Gleichungssysteme:

Quelltext:

```
// Numerik Beleg 4 - Aufgabe 1
// Lösung überbestimmter linearer Gleichungssysteme mit QR-Zerlegung
// Matthias Jauernig, 2004
/* ----- Includes ----- */
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <math.h>

/* ----- Funktions-Deklaration ----- */
void HOUSE(int, int, double**, double*, bool*);

/* ----- main() ----- */
int main(void){
    bool Sing=false;
    int N, M, i, j;
    double **A, *R, *x, RES;

    printf("\n=====
    "\n| Loesung (ueberbestimmter) linearer Gleichungssysteme |"
    "\n=====\\n\\n");

    do{
        printf("Dimension M: ");
        scanf("%d",&M);
    }while(M<2 && printf("Dim. muss groesser gleich 2 sein!\\n\\n"));
    do{
        printf("Dimension N: ");
        scanf("%d",&N);
    }while((N<2 && printf("Dim. muss groesser gleich 2 sein!\\n\\n"))
        || (N>M && printf("M muss groesser gleich N sein!\\n\\n")));
    //Speicher gem. der Dimension allokiieren (M Zeilen, N Spalten)
    A=(double**)malloc(M*sizeof(double*));
    for(i=0; i<M; i++)
        A[i]=(double*)malloc((N+1)*sizeof(double));
    R=(double*)malloc(N*sizeof(double));
    x=(double*)malloc(N*sizeof(double));

    printf(">> Eingabe von Matrix A\\n");
    for(i=0;i<M;i++){
        printf(" - %d. Zeile:\\n",i+1);
        for(j=0;j<N; j++){
            printf(" + %d. Element: ",j+1);
            scanf("%lf",&A[i][j]);
        }
    }

    printf("\\n>> Eingabe von Vektor b\\n");
    for(i=0;i<M;i++){
        printf(" - %d. Element: ",i+1);
        scanf("%lf",&A[i][N]);
    }

    printf("\\n=====\\n"
    "<< Eingegebene Matrix A|b (M=%d, N=%d)\\n",M,N);
    for(i=0;i<M;i++){
        for(j=0;j<=N;j++){
            printf("%11.7lg ", A[i][j]);
        }
        printf("\\n");
    }
}
```

```

HOUSE(M,N,A,R,&Sing); //Householder-Transformationen
for(i=N-1; i>=0; i--){ //Lösungsvektor x berechnen
    x[i]=A[i][N];
    for(j=i+1; j<N; j++)
        x[i]-=A[i][j]*x[j];
    x[i]/=R[i];
}
RES=0.0; //Residuum berechnen
for(i=N; i<M; i++)
    RES+=A[i][N]*A[i][N];
RES=sqrt(RES);

if(Sing){
    printf("Singularitaetstest nicht bestanden - Abbruch\n");
    return 1;
}

printf("\n=====
"<< Ausgabe von Matrix R|c (A|b QR-faktorisiert)\n");
for(i=0;i<M;i++){
    for(j=0;j<=N;j++){
        printf("%11.7lg ", A[i][j]);
    }
    printf("\n");
}

printf("\n<< Ausgabe von Vektor R\n");
for(i=0;i<N;i++)
    printf("%11.7lg\n",R[i]);

printf("\n<< Ausgabe des Loesungsvektors x\n");
for(i=0;i<N;i++)
    printf("%11.7lg\n",x[i]);

printf("\n<< Ausgabe des Residuums (||Ax-b||)\n");
printf("%11.7lg\n",RES);

printf("\n\n");
return 0;
}

/* ----- Funktions-Definition ----- */
void HOUSE(int M, int N, double **A, double *R, bool *Sing){
    int i, j, k;
    double y[N+1], kappa, s;
    for(j=0; j<N; j++){
        //s berechnen
        s=0.0;
        for(i=j; i<M; i++)
            s+=A[i][j]*A[i][j];
        if(s==0.0){ //A singular?
            *Sing=true;
            return;
        }
        //Rho in R abspeichern
        R[j]=(A[j][j]<0)?sqrt(s):-sqrt(s);

        //Kappa berechnen
        kappa=s-R[j]*A[j][j];

        A[j][j]-=R[j];
        for(i=j+1; i<=N; i++){
            y[i]=0; //y berechnen
            for(k=j; k<M; k++)
                y[i]+=A[k][j]*A[k][i];
            y[i]/=kappa;
            for(k=j; k<M; k++) //Householder-transf. Matrix A berechnen
                A[k][i]-=A[k][j]*y[i];
        }
    }
}

```

Testbeispiele:

(a) Ausgabe:

```
<< Eingegebene Matrix A|b (M=4, N=3)
  1      2      3      4
-1      0      1      2
-1     -2     -1      3
  1      0     -1      4

=====
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
  3      -2      -1  5.551115e-17
-1     3.333333      -3      2
-1    -0.6666667    2.814214    -2.828427
  1    -1.333333     -0.2     4.242641

<< Ausgabe von Vektor R
  -2
  -2
-1.414214

<< Ausgabe des Loesungsvektors x
  3
 -4
  2

<< Ausgabe des Residuums (||Ax-b||)
  4.242641
```

(b) Ausgaben:

n=2:

```
<< Eingegebene Matrix A|b (M=12, N=2)
  1      0.5    102.102
  0.5    0.3333333    58.344
  0.3333333    0.25    36.463
  0.25     0.2     24.31
  0.2    0.1666667    17.017
  0.1666667    0.1428571    12.376
  0.1428571     0.125     9.282
  0.125    0.1111111     7.14
  0.1111111     0.1     5.61
  0.1    0.09090909     4.488
  0.09090909    0.08333333    3.6465
  0.08333333    0.07692308     3.003

=====
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
  2.25099    -0.737877    -126.9748
  0.5    0.2209478     7.094684
  0.3333333    0.06669149     2.430223
  0.25    0.06251862    -1.235275
  0.2    0.05668156    -3.430256
  0.1666667    0.05120289    -4.669949
  0.1428571    0.04643921    -5.333036
  0.125    0.04237042    -5.65103
  0.1111111    0.03889716    -5.761848
  0.1    0.03591654    -5.748168
  0.09090909    0.0333401    -5.660248
  0.08333333    0.03109595    -5.52907

<< Ausgabe von Vektor R
  -1.25099
 -0.1625772

<< Ausgabe des Loesungsvektors x
  127.2392
 -43.63885

<< Ausgabe des Residuums (||Ax-b||)
  15.17339
```

n=3:

```
<< Eingegebene Matrix A|b (M=12, N=3)
      1      0.5      0.3333333      102.102
      0.5      0.3333333      0.25      58.344
0.3333333      0.25      0.2      36.463
      0.25      0.2      0.1666667      24.31
      0.2      0.1666667      0.1428571      17.017
0.1666667      0.1428571      0.125      12.376
0.1428571      0.125      0.1111111      9.282
      0.125      0.1111111      0.1      7.14
0.1111111      0.1      0.09090909      5.61
      0.1      0.09090909      0.08333333      4.488
0.09090909      0.08333333      0.07692308      3.6465
0.08333333      0.07692308      0.07142857      3.003
```

```
=====
<< Ausgabe von Matrix R|c (A|b QR-faktorisiert)
      2.25099      -0.737877      -0.5402314      -126.9748
      0.5      0.2209478      -0.1842025      7.094684
0.3333333      0.06669149      -0.01652562      -15.15101
      0.25      0.06251862      0.001691088      0.5638355
      0.2      0.05668156      0.003630249      0.4318835
0.1666667      0.05120289      0.004664236      0.2922263
0.1428571      0.04643921      0.005193302      0.1919996
      0.125      0.04237042      0.005434825      0.1309567
0.1111111      0.03889716      0.005509226      0.09929194
      0.1      0.03591654      0.005485305      0.08752309
0.09090909      0.0333401      0.005403544      0.08845993
0.08333333      0.03109595      0.005288356      0.0970921
```

```
<< Ausgabe von Vektor R
      -1.25099
      -0.1625772
      0.01467426
```

```
<< Ausgabe des Loesungsvektors x
      -116.8913
      1126.187
      -1032.489
```

```
<< Ausgabe des Residuums (||Ax-b||)
      0.8237807
```

n=4:

```
<< Eingegebene Matrix A|b (M=12, N=4)
      1      0.5      0.3333333      0.25      102.102
      0.5      0.3333333      0.25      0.2      58.344
0.3333333      0.25      0.2      0.1666667      36.463
      0.25      0.2      0.1666667      0.1428571      24.31
      0.2      0.1666667      0.1428571      0.125      17.017
0.1666667      0.1428571      0.125      0.1111111      12.376
0.1428571      0.125      0.1111111      0.1      9.282
      0.125      0.1111111      0.1      0.09090909      7.14
0.1111111      0.1      0.09090909      0.08333333      5.61
      0.1      0.09090909      0.08333333      0.07692308      4.488
0.09090909      0.08333333      0.07692308      0.07142857      3.6465
0.08333333      0.07692308      0.07142857      0.06666667      3.003
```

```
=====
<< Ausgabe von Matrix R|c (A|b QR-faktorisiert)
      2.25099      -0.737877      -0.5402314      -0.4312091      -126.9748
      0.5      0.2209478      -0.1842025      -0.181082      7.094684
0.3333333      0.06669149      -0.01652562      0.02441321      -15.15101
      0.25      0.06251862      0.001691088      -0.001786342      -0.7661724
      0.2      0.05668156      0.003630249      -0.0006153688      -0.02628468
0.1666667      0.05120289      0.004664236      -0.0004697372      -0.05751297
0.1428571      0.04643921      0.005193302      -0.0003172468      -0.04420413
      0.125      0.04237042      0.005434825      -0.0001817558      -0.004368201
0.1111111      0.03889716      0.005509226      -6.840058e-05      0.0483648
      0.1      0.03591654      0.005485305      2.397595e-05      0.1053742
0.09090909      0.0333401      0.005403544      9.832673e-05      0.1616684
0.08333333      0.03109595      0.005288356      0.0001577948      0.2145771
```

```
<< Ausgabe von Vektor R
      -1.25099
      -0.1625772
```

```
0.01467426
0.001109488
```

```
<< Ausgabe des Loesungsvektors x
```

```
-60.88815
593.6597
116.3849
-690.5636
```

```
<< Ausgabe des Residuums (||Ax-b||)
```

```
0.3026459
```

n=5:

```
<< Eingegebene Matrix A|b (M=12, N=5)
```

```
1 0.5 0.3333333 0.25 0.2 102.102
0.5 0.3333333 0.25 0.2 0.1666667 58.344
0.3333333 0.25 0.2 0.1666667 0.1428571 36.463
0.25 0.2 0.1666667 0.1428571 0.125 24.31
0.2 0.1666667 0.1428571 0.125 0.1111111 17.017
0.1666667 0.1428571 0.125 0.1111111 0.1 12.376
0.1428571 0.125 0.1111111 0.1 0.09090909 9.282
0.125 0.1111111 0.1 0.09090909 0.08333333 7.14
0.1111111 0.1 0.09090909 0.08333333 0.07692308 5.61
0.1 0.09090909 0.08333333 0.07692308 0.07142857 4.488
0.09090909 0.08333333 0.07692308 0.07142857 0.06666667 3.6465
0.08333333 0.07692308 0.07142857 0.06666667 0.0625 3.003
```

```
=====
```

```
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
```

```
2.25099 -0.737877 -0.5402314 -0.4312091 -0.3608771 -126.9748
0.5 0.2209478 -0.1842025 -0.181082 -0.171999 7.094684
0.3333333 0.06669149 -0.01652562 0.02441321 0.03022633 -15.15101
0.25 0.06251862 0.001691088 -0.001786342 0.002418278 -0.7661724
0.2 0.05668156 0.003630249 -0.0006153688 -7.938336e-05 0.3015261
0.1666667 0.05120289 0.004664236 -0.0004697372 -1.251819e-05 -0.005819562
0.1428571 0.04643921 0.005193302 -0.0003172468 -7.699666e-06 -0.01240863
0.125 0.04237042 0.005434825 -0.0001817558 2.442934e-06 -0.01445621
0.1111111 0.03889716 0.005509226 -6.840058e-05 1.468206e-05 -0.01226422
0.1 0.03591654 0.005485305 2.397595e-05 2.71928e-05 -0.006917492
0.09090909 0.0333401 0.005403544 9.832673e-05 3.903512e-05 0.0004742164
0.08333333 0.03109595 0.005288356 0.0001577948 4.977947e-05 0.009014478
```

```
<< Ausgabe von Vektor R
```

```
-1.25099
-0.1625772
0.01467426
0.001109488
7.231001e-05
```

```
<< Ausgabe des Loesungsvektors x
```

```
44.46694
-1095.013
6648.057
-9779.435
4169.908
```

```
<< Ausgabe des Residuums (||Ax-b||)
```

```
0.026011
```

n=6:

```
<< Eingebene Matrix A|b (M=12, N=6)
  1      0.5      0.3333333      0.25      0.2      0.1666667      102.102
  0.5      0.3333333      0.25      0.2      0.1666667      0.1428571      58.344
  0.3333333      0.25      0.2      0.1666667      0.1428571      0.125      36.463
  0.25      0.2      0.1666667      0.1428571      0.125      0.1111111      24.31
  0.2      0.1666667      0.1428571      0.125      0.1111111      0.1      17.017
  0.1666667      0.1428571      0.125      0.1111111      0.1      0.09090909      12.376
  0.1428571      0.125      0.1111111      0.1      0.09090909      0.08333333      9.282
  0.125      0.1111111      0.1      0.09090909      0.08333333      0.07692308      7.14
  0.1111111      0.1      0.09090909      0.08333333      0.07692308      0.07142857      5.61
  0.1      0.09090909      0.08333333      0.07692308      0.07142857      0.06666667      4.488
  0.09090909      0.08333333      0.07692308      0.07142857      0.06666667      0.0625      3.6465
  0.08333333      0.07692308      0.07142857      0.06666667      0.0625      0.05882353      3.003

=====
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
  2.25099      -0.737877      -0.5402314      -0.4312091      -0.3608771      -0.311272      -126.9748
  0.5      0.2209478      -0.1842025      -0.181082      -0.171999      -0.161758      7.094684
  0.3333333      0.06669149      -0.01652562      0.02441321      0.03022633      0.03356618      -15.15101
  0.25      0.06251862      0.001691088      -0.001786342      0.002418278      0.003601909      -0.7661724
  0.2      0.05668156      0.003630249      -0.000615369      -7.938336e-05      0.0001940279      0.3015261
  0.1666667      0.05120289      0.004664236      -0.000469737      -1.251819e-05      -4.95489e-06      0.02600439
  0.1428571      0.04643921      0.005193302      -0.000317247      -7.699666e-06      -1.929384e-06      -1.67376e-05
  0.125      0.04237042      0.005434825      -0.000181756      2.442934e-06      -2.269602e-06      0.0001208434
  0.1111111      0.03889716      0.005509226      -6.84006e-05      1.468206e-05      -1.927783e-06      0.0001174188
  0.1      0.03591654      0.005485305      2.397595e-05      2.71928e-05      -1.075178e-06      -1.19084e-05
  0.09090909      0.0333401      0.005403544      9.832673e-05      3.903512e-05      1.098309e-07      -0.000231199
  0.08333333      0.03109595      0.005288356      0.0001577948      4.977947e-05      1.483134e-06      -0.0005113

<< Ausgabe von Vektor R
  -1.25099
  -0.1625772
  0.01467426
  0.001109488
  7.231001e-05
  4.087745e-06

<< Ausgabe des Loesungsvektors x
  -2.157811
  49.13948
  -282.2486
  6773.978
  -12899.9
  6361.549

<< Ausgabe des Residuums (||Ax-b||)
  0.0005862521
```

2.) Ellipsenbahn eines Himmelskörpers um die Sonne:

Ausgabe des Programms aus 1.:

```
<< Eingegebene Matrix A|b (M=10, N=5)
 0.1586182  0.4082018  -1.02494  -0.398269  1  1.050502
 0.1042605  0.3067164  -0.949898  -0.322894  1  0.9023062
 0.07036075  0.2297419  -0.866114  -0.265256  1  0.7501535
 0.04689693  0.1674835  -0.773392  -0.216557  1  0.5971352
 0.03138283  0.1189349  -0.671372  -0.177152  1  0.4507404
 0.02178045  0.08257567  -0.559524  -0.147582  1  0.3130671
 0.01654259  0.05621468  -0.437067  -0.128618  1  0.1910276
 0.01472655  0.03675892  -0.302909  -0.121353  1  0.09175386
 0.01621751  0.01980172  -0.155493  -0.127348  1  0.02417807
 0.02216674  0.001111278  -0.007464  -0.148885  1  5.57113e-05

=====
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
 0.372824  -0.5979951  1.8975  0.6709042  -2.34799  -1.714976
 0.1042605  0.1243603  0.6906439  0.05309773  -0.236844  -0.4688023
 0.07036075  0.03984866  -0.5933848  0.2171517  -2.081221  -0.1109891
 0.04689693  0.04091555  -0.1349174  0.0550007  0.2637519  0.06152588
 0.03138283  0.03423726  -0.1987182  0.01976179  -0.2106731  -0.08156312
 0.02178045  0.02379346  -0.2312792  0.02937339  2.063942e-05  0.002283297
 0.01654259  0.01156867  -0.2308094  0.02554491  0.05627631  0.001920014
 0.01472655  -0.00298587  -0.2072394  0.008423672  0.1019906  0.0005754364
 0.01621751  -0.02396695  -0.1870333  -0.01500175  0.08366048  0.0009477523
 0.02216674  -0.05871349  -0.2223962  -0.02573689  -0.08823717  0.0003925623

<< Ausgabe von Vektor R
-0.2142058
-0.09902774
 0.5426073
-0.05353879
 0.1726303

<< Ausgabe des Loesungsvektors x
-2.234505
-0.3615626
-0.6253605
-3.476758
-0.4724727

<< Ausgabe des Residuums (||Ax-b||)
0.003206769
```

Die Lösung des linearen Ausgleichsproblems ist daher:

- a = -2.234505
- b = -0.3615626
- c = -0.6253605
- d = -3.476758
- e = -0.4724727

3.) Sauerstoff-Stickstoff-Verbindungen:

Ausgabe des Programms aus 1.:

```
<< Eingegebene Matrix A|b (M=6, N=2)
      1      1      30.006
      2      1      44.013
      1      2      46.006
      2      3      76.012
      2      5      108.01
      2      4      92.011

=====
<< Ausgabe von Matrix R|c (A|b QR-faktoriert)
 5.242641  -6.835366  -168.7873
      2  -5.035036   48.73296
      1  0.5054544  0.0005323848
      2  0.01090881 0.0003672783
      2   2.010909 -0.000237739
      2   1.010909 6.476966e-05

<< Ausgabe von Vektor R
-4.242641
 3.045944

<< Ausgabe des Lösungsvektors x
 14.00692
 15.99929

<< Ausgabe des Residuums (||Ax-b||)
0.0006921285
```

Daher ergibt sich als Lösung für die Atomgewichte:

$$N = 14.00692$$

$$O = 15.99929$$