Lucrarea de laborator nr. 5

I. Scopul lucrării

Aplicații ale eliminării gaussiene cu pivotare parțială:
- calculul determinantului unei matrice
- rezolvarea sistemelor liniare
- calculul inversei unei matrice

II. Conținutul lucrării

1. Prezentarea metodei de eliminare Gauss cu pivotare parțială
2. Calculul determinantului unei matrice – procedura MAPLE.
3. Rezolvarea sistemelor liniare compatibil determinate cu n ecuații şi n necunoscute - procedura MAPLE.
4. Calculul inversei unei matrice - procedura MAPLE.

III. Prezentarea lucrării

III. 1. Prezentarea metodei de eliminare Gauss cu pivotare parțială

Se consideră o matrice $A \in M_{n,m}(\mathbf{R})$. ***Eliminarea gaussiană*** urmăreşte transformarea matricei A într-o matrice superior triunghiulară B (o matrice cu proprietatea că $b_{ij} = 0$ pentru orice $i<j$). Trecerea de la matricea A la matricea B se realizează prin transformări elementare. La baza metodei stă următorul procedeu:

- prima linie este folosită pentru anularea coeficienților de pe prima coloană din celelalte n-1 linii.
- a doua linie este utilizată pentru anularea coeficienților de pe a doua coloană din ultimele n-2 linii, ş.a.m.d.

Trecerea de la un pas la altul se face aplicând regula dreptunghiului (pivotului). Pentru a obține stabilitatea numerică a algoritmului, se alege drept pivot de la pasul k elementul maxim în modul din coloana k subdiagonală a lui A, şi se permută linia k cu linia pe care se găseşte pivotul. Această strategie de permutare se numeşte ***pivotare parţială***. Performanţe de stabilitate numerică relativ mai bune se obţin dacă se alege drept pivotul la pasul k elementul maxim în modul din submatricea delimitată de ultimele n-k linii, şi se permută coloana k cu coloana pivotului şi linia k cu linia pivotului. Această strategie de pivotare se numeşte ***pivotare completă***. Prezentăm în continuare algoritmul de eliminare Gauss cu pivotare parţială. Trecerea de la matricea A la matricea superior triunghiulară se realizează în nmin paşi, unde nmin=min{n,m}:

$$A^{(0)} \to \quad A^{(1)} \to \ldots \to A^{(nmin)},$$

unde $A^{(nmin)}$ are formă superior triunghiulară, iar $A^{(0)} = A$.

Pentru a se trece de la $A^{(k)} \to A^{(k+1)}$:

- Se determină pivotul de la pasul k; acesta este primul element $a_{i,k}$ de pe coloana k cu proprietatea

$$|a_{i,k}| = \max\{|a_{j,k}|,\ k \le j \le n\}$$

- Se permută liniile i cu k;
- Se aplică regula dreptunghiului (pivotului) cu pivotul $a_{k,k}$. Astfel:

- elementele de pe linia pivotului se împart la pivot:

$$a_{k,i}^{(k+1)} = a_{ki}^{(k)} / a_{kk}^{(k)}, \ i=k,k+1,\ldots m$$

- elementele de pe coloana pivotului, cu excepţia pivotului, se înlocuiesc cu 0.

$$a_{i\,k}=0,\ i= k+1,\ k+2,\ldots n$$

- elementele din submatricea delimitată de ultimele n-k linii se transformă cu regula dreptunghiului:

$$\begin{array}{ccc} a_{kk}^{(k)} & & a_{kj}^{(k)} \\ & & \\ a_{ik}^{(k)} & & a_{ij}^{(k)} \end{array}$$

$$a_{ij}^{(k+1)} = \frac{a_{ij}^{(k)} a_{kk}^{(k)} - a_{kj}^{(k)} a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

$$k+1 \le i \le n,\ k+1 \le j \le m$$

În urma aplicării acestui algoritm se ajunge la următoarea matrice superior triunghiulară:

$$A^{(nmin)} = \begin{pmatrix} 1 & a_{1,2}^{(1)} & a_{1,3}^{(1)} \cdots\cdots & a_{1,n}^{(1)} & a_{1,n+1}^{(1)} & \cdots\cdots & a_{1,m}^{(1)} \\ 0 & 1\cdot & a_{2,3}^{(2)} \cdots\cdots & a_{2,n}^{(2)} & a_{2,n+1}^{(2)} & \cdots\cdots & a_{2,m}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 \cdots\cdots & 1 & a_{n,n+1}^{(n)} & \cdots\cdots & a_{n,m}^{(n)} \end{pmatrix}$$

(dacă m≤n).

Următoarea procedură MAPLE are drept parametri matricea B, numărul de linii n, numărul de coloane m, şi întoarce matricea superior triunghiulară obținută prin aplicarea algoritmului de eliminare Gauss cu pivotare parțială asupra matricei B. Matricea A se inițializează cu B, şi fiecare din matricele intermediare $A^{(k)}$ sunt afişate. Pentru memorarea acestora se foloseşte aceeaşi matrice A. Modificarea coeficienților de la pasul k la pasul k+1 este dată de formulele:

$$a_{k,i} := \frac{a_{k,i}}{a_{k,k}},\ i = k+1,\ldots,m;\ a_{k,k} := 1.$$

$$a_{i,j} := a_{i,j} - a_{i,k}\, a_{k,j},\ i=k+1,\ldots,n,\ j=k+1,\ldots,m.$$

$$a_{i,k} := 0,\ i = k+1,\ldots,n.$$

```
> elimpp := proc(b, n, m)
      local a, x, i, j, k, aux;
      a := b;
      if n<=m then nmin:=n else nmin:=m fi;
      for k to nmin do
           aux := abs(a[k, k]);
           i := k;
           for j from k + 1 to n do
            if aux < abs(a[j, k]) then
         aux := abs(a[j, k]);
         i := j
      fi
   od;
   for j from k to m do
      aux := a[k, j];
      a[k, j] := a[i, j];
      a[i, j] := aux
   od;
```

```
   if a[k, k] <> 0 then
   for j from k + 1 to m do
      a[k, j] := a[k, j]/a[k, k]
   od;
   a[k, k] := 1;
   for i from k + 1 to n do
    for j from k + 1 to m do
        a[i, j] := a[i, j] - a[k, j]*a[i, k]
      od
   od;
   for i from k + 1 to n do
     a[i, k] := 0
   od;
   print(a)
  fi;
od;
print(`Matricea superior triunghiulara obtinuta`);
RETURN(evalm(a))
end;
```

```
>A:=matrix(3,3,[[1,1,1],[1,2,3],[2,1,2]]);
```

$$A := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 2 & 1 & 2 \end{pmatrix}$$

```
> elimpp(A,3,3);
```

$$\begin{pmatrix} 1 & 1/2 & 1 \\ 0 & 3/2 & 2 \\ 0 & 1/2 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1/2 & 1 \\ 0 & 1 & 4/3 \\ 0 & 0 & -2/3 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 1/2 & 1 \\ 0 & 1 & 4/3 \\ 0 & 0 & 1 \end{pmatrix}$$

```
Matricea superior triunghiulara obtinuta
```

$$\begin{pmatrix} 1 & 1/2 & 1 \\ 0 & 1 & 4/3 \\ 0 & 0 & 1 \end{pmatrix}$$

Comanda **matrix** face parte din pachetul **linalg**, deci înainte de utilizare trebuie încărcat acest pachet, printr-o comanda with(linalg), sau alternativ trebuie specificat pachetul înaintea comenzii. Comanda **gausselim** din acelaşi pachet aplică algoritmul de eliminare Gauss (fără pivotare) asupra matricei de intrare. În exemplul următor presupunem că A este matricea definită anterior.

```
>gausselim(A);
```

$$\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 2 \end{pmatrix}$$

III.2. Calculul determinantului unei matrice – procedura MAPLE.

Pentru calculul determinantului unei matrice A se poate folosi metoda de eliminare Gauss, ţinându-se seama că determinantul unei matrice nu se modifică dacă asupra matricei se efectuează transformări elementare. Determinantul este produsul elementelor $a_{k,k}^{(k)}$ (înainte de împărţire). Dacă se foloseşte pivotare, trebuie avut în vedere faptul că prin permutarea a două linii semnul determinantului se schimbă. Procedura determinant de mai jos are drept parametri matricea B şi numărul de linii ale ei, şi întoarce determinantul matricei. Pentru transformarea matricei B într-o matrice superior triunghiulară se foloseşte algoritmul de eliminare Gauss cu pivotare parţială.

```
>determinant := proc(b, n)
  local a, d, i, j, k, aux;
   a := b;
   d := 1;
   for k to n do
```

```
    aux := abs(a[k, k]);
    i := k;
    for j from k + 1 to n - 1 do
     if aux < abs(a[j, k]) then
         aux := abs(a[j, k]); i := j
        fi
       od;
       if k <> i then
        for j from k to n do
         aux := a[k, j];
         a[k, j] := a[i, j];
         a[i, j] := aux
        od;
        d := -d
       fi;
       d := d*a[k, k];
       if a[k, k] = 0 then RETURN(0) fi;
       for j from k + 1 to n do
                a[k, j] := a[k, j]/a[k, k]
       od;
       a[k, k] := 1;
       for i from k + 1 to n do
        for j from k + 1 to n do
        a[i, j] := a[i, j] - a[k, j]*a[i, k]
        od
       od;
       for i from k + 1 to n do
         a[i, k] := 0
       od
     od;
      d := d*a[n, n];
      RETURN(d)
end;
```

```
>A:=matrix(3,3,[[1,1,1],[1,8,3],[12,1,2]]);
```

$$A := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 28 & 3 \\ 12 & 1 & 2 \end{pmatrix}$$

```
> determinant(A,3);
```

```
                          -48

>B:=matrix(4,4,[6,5,8,3,90,1,2,3,4,5,7,4,8,2,1,9]);
```

$$B := \begin{bmatrix} 6 & 5 & 8 & 3 \\ 90 & 1 & 2 & 3 \\ 4 & 5 & 7 & 4 \\ 8 & 2 & 1 & 9 \end{bmatrix}$$

```
> determinant(B,4);
                          2324
```

Pentru calculul determinantului unei matrice se poate folosi şi comanda **det** din pachetul **linalg**. În exemplele următoare A şi B sunt matricele definite mai sus.

```
> det(A);
                          -48
> det(B);
                          2324
```

III.3. Rezolvarea sistemelor liniare compatibil determinate cu n ecuații şi n necunoscute - procedura MAPLE.

Considerăm sistemul cu n ecuații şi n necunoscute.

$$Ax = b,\ A \in M_{n,n}(\mathbf{R}) \text{ nesingulară}$$

Pentru rezolvarea acestui sistem vom aplica algoritmul de eliminare Gauss cu pivotarea parțială asupra matricei extinse $\overline{A} = (A \mid b)$. Vom nota elementele matricei $\overline{A}$ tot cu $a_{i,j}$. Astfel $a_{i,n+1}= b_i$ pentru orice i=1,2,..n.

La primul pas algoritmul presupune eliminarea necunoscutei x_1 din ultimele n-1 ecuații. La al doilea pas se elimină necunoscuta x_2 din ultimele n-2 ecuații, ş.a.m.d. Ca urmare a aplicării algoritmului se obține sistemul echivalent:

$$\begin{cases} x_1 + a^{(1)}_{1,2}x_2 + a^{(1)}_{1,3}x_3 + \cdots\cdots\cdots\cdots a^{(1)}_{1,n}x_n = a^{(1)}_{1,n+1} \\ \quad x_2 + a^{(2)}_{2,3}x_3 + \cdots\cdots\cdots\cdots a^{(2)}_{2,n}x_n = a^{(2)}_{2,n+1} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \qquad\qquad\qquad x_n = a^{(n)}_{n,n+1} \end{cases}$$

Rezolvarea acestui sistem se poate face foarte uşor de la sfârşit spre început:

$$x_n = a^{(n)}_{n,n+1}$$

$$x_i = a^{(i)}_{i,n+1} - \sum_{j=i+1}^{n} a^{(i)}_{i,j}x_j$$

Procedura rezsist de mai jos întoarce soluția unui sistem liniar cu n ecuații şi n necunoscute. Parametri procedurii sunt numărul de ecuații şi matricea extinsă a sistemului.

```
>rezsist := proc(n, s)
  local a, x, i, j, k, aux;
  x := vector(n);
  for i to n do
     for j to n+1 do
         a[i,j]:= s[i,j]
      od
   od
  for k to n do
  aux := abs(a[k, k]);
  i := k;
  for j from k + 1 to n do
  if aux < abs(a[j, k]) then
   aux := abs(a[j, k]);
   i := j
  fi
 od;
 for j from k to n + 1 do
  aux := a[k, j];
  a[k, j] := a[i, j];
  a[i, j] := aux
 od;
```

```
if a[k, k] = 0 then
 print(`Matrice singulara`);
 print(a); RETURN
fi;
for j from k + 1 to n + 1 do
 a[k, j] := a[k, j]/a[k, k]
od;
a[k, k] := 1;
for i from k + 1 to n do
 for j from k + 1 to n + 1 do
       a[i, j] := a[i, j] - a[k, j]*a[i, k]
 od
od;
for i from k + 1 to n do
  a[i, k] := 0 od
od;
x[n] := a[n, n + 1];
for i from n - 1 by -1 to 1 do
 x[i] := a[i, n + 1];
 for j from i + 1 to n do
  x[i] := x[i] - a[i, j]*x[j]
 od
od;
RETURN(evalm(x))
end;
```

Utilizăm procedura se mai sus pentru rezolvarea sistemului

$$\begin{cases} x_1 + x_2 + x_3 = 3 \\ x_1 + 2x_2 + 3x_3 = 6 \\ 2x_1 + x_2 + 2x_3 = 6 \end{cases}$$

```
>S:=matrix(3,4,[[1,1,1,3],[1,2,3,6],[2,1,2,6]]);
```

$$S := \begin{pmatrix} 1 & 1 & 1 & 3 \\ 1 & 2 & 3 & 6 \\ 2 & 1 & 2 & 6 \end{pmatrix}$$

```
> rezsist(3,S);
                    [3/2, 0, 3/2]
```

Deci soluția sistemului este:

$x_1=3/2$

$x_2=0$
$x_3=3/2$.

Probleme propuse

1) Considerăm sistemul Ax = b, unde

$$A = \begin{pmatrix} 888.445 & 887.112 \\ 887.112 & 885.781 \end{pmatrix}$$

şi

$$b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Soluția corectă a sistemului este

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 885.781 \\ -887.112 \end{pmatrix}$$

Utilizând procedura rezsist rezultatul obținut este diferit. Explicați! Indicație: calculați factorul de condiționare al matricei A, cond(A) = $\|A\|\|A^{-1}\|$ şi demonstrați astfel că sistemul este rău condiționat

2) Considerăm sistemul Ax = b, unde

$$A = \begin{pmatrix} 1 & 1.001 \\ 1 & 1 \end{pmatrix}$$

şi

$$b = \begin{pmatrix} 2.001 \\ 2 \end{pmatrix}$$

Evident soluția corectă a sistemului este

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Dacă se consideră vectorul y

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}$$

se observă că

$$b - Ay = \begin{pmatrix} 0.001 \\ 0 \end{pmatrix}$$

Deci y aparent verifică sistemul deşi este diferit de x. De ce? Utilizaţi relaţia

$$\frac{\left\|x^{*}-x\right\|}{\|x\|} \leq \operatorname{cond}(A) \frac{\left\|b-Ax^{*}\right\|}{\|b\|}$$

unde x este soluţia exactă a sistemului, iar x* este soluţia aproximativă calculată.

III.4. Calculul inversei unei matrice - procedura MAPLE.

Dacă $A \in M_{n,n}$ (**R**). A inversabilă dacă şi numai dacă există o matrice $B \in M_{n,n}$ (**R**) astfel încât $A \cdot B = B \cdot A = I_n$, unde

$$I_n = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

Se ştie că A este inversabilă dacă şi numai dacă are determinantul nenul. Notăm:

$$e_k = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \\ \\ \\ \leftarrow k \\ \\ \\ \\ \end{matrix}$$

vectorul k al bazei canonice din $\mathbf{R}^n$.

Pentru fiecare k (k=1,2,…,n) considerăm sistemul $Ax=e_k$. Soluţia acestui sistem reprezintă chiar coloana k a matricei A^{-1}. Astfel pentru aflarea lui A^{-1} este necesar să rezolvăm n sisteme de ecuaţii liniare $Ax=e_k$, k=1,2,…,n. Aceste sisteme pot fi rezolvate utilizând algoritmul de eliminare al lui Gauss. Pentru a micşora volumul de calcul vom aplica algoritmul asupra matricei obținute prin concatenarea la matricea A a coloanelor bazei canonice din $\mathbf{R}^n$. Coeficienții matricei

$$\overline{A} = (A/e_1/e_2\text{------}e_n)$$

sunt

a_{ij}, dacă $1 \le i, j \le n$

$a_{i,n+k}=\delta_{ik}$, dacă $1 \le i, k \le n$

unde

$$\delta_{ik} = \begin{cases} 1, & i=k \\ 0, & i \ne k. \end{cases}$$

Prin aplicarea algoritmului de eliminare Gauss cu pivotare parțială asupra lui $\overline{A}$ se obține un şir de matrice

$$A^{(0)} \to \quad A^{(1)} \to \quad \text{———}\to A^{(n)},$$

unde $A^{(0)} = \overline{A}$, iar $A^{(n)}$ are următoarea formă superior triunghiulară:

$$A^{(n)} = \begin{pmatrix} 1 & a_{1,2}^{(1)} & a_{1,3}^{(1)} \cdots\cdots & a_{1,n}^{(1)} & a_{1,n+1}^{(1)} & \cdots\cdots & a_{1,2n}^{(1)} \\ 0 & 1\cdot & a_{2,3}^{(2)} \cdots\cdots & a_{2,n}^{(2)} & a_{2,n+1}^{(2)} & \cdots\cdots & a_{2,2n}^{(2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 \cdots\cdots & 1 & a_{n,n+1}^{(n)} & \cdots\cdots & a_{n,2n}^{(n)} \end{pmatrix}$$

Dacă $B=A^{-1}$, atunci coeficienții lui B pot fi determinați cu formulele:

$$\begin{cases} b_{n,k} = a_{n,n+k}^{(n)}, \ 1 \le k \le n \\ b_{i,k} = a_{i,n+k}^{(i)} - \sum_{j=i+1}^{n} a_{i,j}^{(i)} b_{i,j}, \ 1 \le i \le n-1, \ 1 \le k \le n \end{cases}$$

Procedura inversa de mai jos întoarce inversa matricei indicată drept al doilea parametru. Primul parametru reprezintă numărul de linii al matricei.

```
>inversa := proc(n, s)
  local a, b, i, j, k, aux;
    b := matrix(n, n);
    a := matrix(n, 2*n);
    for i to n do
```

```
    for j to n do
     a[i, j] := s[i, j]
  od
 od;
 for i to n do
  for j to n do
   a[i, j + n] := 0
  od
 od;
 for i to n do
  a[i, i + n] := 1
 od;
 for k to n do
  aux := abs(a[k, k]);
  i := k;
 for j from k + 1 to n do
  if aux < abs(a[j, k]) then
   aux := abs(a[j, k]); i := j
  fi
 od;
 for j from k to 2*n do
  aux := a[k, j];
  a[k, j] := a[i, j];
  a[i, j] := aux
 od;
 if a[k, k] = 0 then
  print(`Matrice singulara`);
  print(a); RETURN
 fi;
 for j from k + 1 to 2*n do
  a[k, j] := a[k, j]/a[k, k]
 od;
  a[k, k] := 1;
 for i from k + 1 to n do
  for j from k + 1 to 2*n do
       a[i, j] := a[i, j] - a[k, j]*a[i, k]
   od
  od;
  for i from k + 1 to n do
   a[i, k] := 0
  od
 od;
 for i to n do
```

```
    b[n, i] := a[n, i + n]
   od;
   for k to n do
    for i from n - 1 by -1 to 1 do
       b[i, k] := a[i, n + k];
       for j from i + 1 to n do
        b[i, k] := b[i, k] -
                 a[i, j]*b[j, k]
       od
     od
    od;
    RETURN(evalm(b))
end;
```

Calculul inversei matricei

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 2 & 1 & 2 \end{pmatrix}$$

se realizează cu ajutorul comenzilor

```
>A:=matrix(3,3,[[1,1,1],[1,2,3],[2,1,2]]);
```

$$A := \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 2 & 1 & 2 \end{pmatrix}$$

```
> inversa(3,A);
```

$$\begin{pmatrix} 1/2 & -1/2 & 1/2 \\ 2 & 0 & -1 \\ -3/2 & 1/2 & 1/2 \end{pmatrix}$$

Comanda **inverse** din pachetul **linalg** calculează de asemenea inversa unei matrice. În exemplul următor presupunem că o aplicăm asupra aceleaşi matrice A.

```
> inverse(A);
```

$$\begin{pmatrix} 1/2 & -1/2 & 1/2 \\ 2 & 0 & -1 \\ -3/2 & 1/2 & 1/2 \end{pmatrix}$$