

```

algol,n<
begin
  comment

    buffer N=60 no for:
    Time classic:  408.109
    Time turbo:   375.170 8.1pct
;
integer procedure LEQ1(N, M, A, eps);
value N, M, eps;
integer N, M;
array A;
real eps;
begin
  integer i, j, k, i1, j1;
  real max, f2, factor;
  integer array p[1:N];
  M := N + M;
  LEQ1 := 0;
  for i := 1 step 1 until N do
  begin
    max := 0;
    for j := 1 step 1 until N do
      max := max + A[i,j]2;
      if max > 1 ∨ max < 0.25 then
      begin
        f2 := 2⌊(-entier(ln(max)/1.3863 + 1));
        for j := 1 step 1 until M do
          A[i,j] := A[i,j]×f2
        end if max
      end for i: equilibration;
    for k := 1 step 1 until N do
    begin
      max := 0;
      for i := k step 1 until N do
      for j := k step 1 until N do
      begin
        factor := abs(A[i,j]);
        if max ≤ factor then
        begin
          max := factor;
          i1 := i;
          j1 := j
        end if larger
      end;
      if max < eps then
      begin
        LEQ1 := 1;
        go to EX
      end error exit;
      max := A[i1,j1];
      if i1 ≠ k then
      for j := k step 1 until M do
      begin
        factor := A[k,j];
        A[k,j] := A[i1,j];
        A[i1,j] := factor
      end for j: row interchange;
      p[k] := k;
      if j1 ≠ k then
      begin
        p[k] := j1;
        for i := 1 step 1 until N do
        begin

```

```

        factor := A[i,k];
        A[i,k] := A[i,j1];
        A[i,j1] := factor
    end for i
end interchange of columns;
for i := k + 1 step 1 until N do
begin
    factor := A[i,k]/max;
    for j := k + 1 step 1 until M do
        A[i,j] := A[i,j] - A[k,j]×factor
    end
end for k;
for k := N + 1 step 1 until M do
for i := N step -1 until 1 do
begin
    factor := A[i,k];
    for j := i + 1 step 1 until N do
        factor := factor - A[i,j]×A[j,k];
        A[i,k] := factor/A[i,i]
    end solving;
    if M ≠ N then
    for i := N - 1 step -1 until 1 do
    begin
        i1 := p[i];
        if i1 ≠ i then
        for k := N + 1 step 1 until M do
        begin
            factor := A[i,k];
            A[i,k] := A[i1,k];
            A[i1,k] := factor
        end for k
        end for i and solution interchange;
EX:end LEQ1;
real procedure clock count;
code clock count;
1, 37;
z1      , grf p-1    ; RF:=clock count; stack[p-1]:=RF;
e;
real procedure RANDOM;
begin
    integer new,mod;
    mod := 2796203;
    new := 125×oldrand;
    oldrand := new - mod×entier(new/mod);
    RANDOM := oldrand/mod-0.5
end RANDOM;
integer oldrand,N;
real time,maxerror;

select (17);
oldrand:=100001;
writecr;
writetext (⟨<N: ⟩);
N:=60;
writeinteger (⟨p⟩,N);
begin
    array A[1:N,1:N+1];
    integer i,j;
    real sum;
    for i:=1 step 1 until N do
    begin
        sum:=0;
        for j:=1 step 1 until N do
        begin

```

```

        A[i, j] := RANDOM;
        sum := sum + A[i, j]
    end;
    A[i, N+1] := sum
end;
writecr;
clock count;
if LEQ1(N, 1, A, 110-12) = 1 then
    writetext({<Error.>});
    time := clock count;
    write({dddd.d}, time);
    writecr;
    maxerror := 0;
    for i := 1 step 1 until N do
        begin
            sum := abs(A[i, N+1] - 1);
            if sum > maxerror then maxerror := sum
        end for;
        write({d.d10-dd}, maxerror)
    end inner block
end
t <

```