### Задание! Объяснить что происходит в алгоритме решения

begin

s1 := 0;

s2 := 0;

For j := 1 to i - 1 do

s1 := s1 + a[i, j] \* x[j]; {алгоритм решения}

For j := i to n do

s2 := s2 + a[i, j] \* x[j];

x[i]:=s1+s2+b[i];

end;

Что такое S1 и S2, Зачем якобы 2 раза выполнять одно и то же действие?

s1 := s1 + a[i, j] \* x[j];

s2 := s2 + a[i, j] \* x[j];

И что же мы в итоге получим на выходе цикла?

В общем нужна вся информация по алгоритму решения, но и по задаче тоже не наврядит.

### (метод Зейделя)

program seidel;

var A: array [1..4,1..4] of real;

b,x,otv: array [1..4] of real;

i,j,n: byte;

eps,s1,s2: real;

pr: boolean;

begin

write('razmer matrix n=');

readln(n);

for i:=1 to n do

for j:=1 to n do {Ввод данных}

begin

write('A[',i,',',j,']=');

readln(A[i,j]);

end;

for i:=1 to n do

if a[i,i]=0 then begin

writeln('oshibka vvoda'); {Проверка на сходимость}

exit;

end;

for i:=1 to n do

begin

write('b[',i,']=');

readln(b[i]);

end;

for i:=1 to n do

begin

for j:=1 to n do

begin

if i=j then continue;

a[i,j]:=-a[i,j]/a[i,i]; {Выражение аргументов}

end;

b[i]:=b[i]/a[i,i];

a[i,i]:=0;

end;

for i:=1 to n do

begin

for j:=1 to n do

write(a[i,j]:4:2,' ');

writeln(b[i]:4:2);

end;

for i:=1 to n do

begin

x[i]:=b[i];

otv[i]:=b[i];

end;

write('tochnost=');

readln(eps);

repeat

for i:=1 to n do

begin

s1 := 0;

s2 := 0;

For j := 1 to i - 1 do

s1 := s1 + a[i, j] \* x[j]; {алгоритм решения}

For j := i to n do

s2 := s2 + a[i, j] \* x[j];

x[i]:=s1+s2+b[i];

end;

for i:=1 to n do

if abs(otv[i]-x[i])<eps then pr:=true

else begin

pr:=false;

break;

end;

for i:=1 to n do

otv[i]:=x[i];

until pr;

for i:=1 to n do

writeln(x[i]); {Вывод результата}

end.

Контрольный тест

Размер матрицы

4

Введите точность вычислений

0.000001

Введите расширенную матрицу системы

A 1 2 3 4 b

1 4.1 0.1 0.2 0.2 21.14

2 0.3 5.3 0.9 -0.1 -17.82

3 0.2 0.3 3.2 0.2 9.02

4 0.1 0.1 0.2 -9.1 17.08

Результат вычислений по методу Зейделя

5.2000000061E+00

-4.2000000026E+00

2.9999999999E+00

-1.8000000000E+00