Лабораторная работа № 10

Решение системы линейных алгебраических уравнений (СЛАУ) методом Гаусса.

Цель работы: изучить метод Гаусса для решения системы линейных алгебраических уравнений. Написать программу на Си по поиску решения СЛАУ методом Гаусса.

Краткие теоретические сведения Методом Гаусса называют точный метод решения невырожденной системы линейных уравнений, состоящий в том, что последовательным исключением неизвестных систему: приводят к эквивалентной системе с треугольной матрицей: решение которой находят по рекуррентным формулам: Таким образом, вычисления по методу Гаусса распадаются на два этапа. На первом этапе, называемом прямым ходом метода, исходную систему при помощи элементарных преобразований преобразуют к треугольному виду . На втором этапе, который называют обратным ходом, решают треугольную систему , эквивалентную исходной. Отметим, что к элементарным преобразованиям системы относят следующие:

• перестановка любых двух уравнений системы;

• умножение любого уравнения системы на отличное от нуля число;

• вычеркивание уравнения, все коэффициенты которого равны нулю;

• прибавление к одному уравнению системы любого другого, умноженного на отличное от нуля число.

Каждому элементарному преобразованию системы соответствует аналогичное элементарное преобразование над строками расширенной матрицы этой системы. Поэтому на практике элементарным преобразованиям подвергают не систему, а ее расширенную матрицу. Пример решения. Методом Гаусса решить СЛАУ с точностью ε =10-2 . Выпишем расширенную матрицу данной системы: Совершая над строками расширенной матрицы элементарные преобразования, приведем её к специальному виду: По полученной матрице запишем систему уравнений: эквивалентную системе. Закончен прямой ход метода Гаусса. Переходим к обратному ходу.

Составить программу на Си по решению систем:

|  |
| --- |
| #include<stdio.h> |
|  | int main(){ |
|  | int i,j,k,n; |
|  | float A[20][20] , c , x[10],sum=0; |
|  | printf("enter the order of matrix: "); |
|  | scanf("%d",&n); |
|  | printf("/nEnter the elements of argumented matrix"); |
|  | for(i=1;i<=n;i++){ |
|  | for(j=1; j<=(n+1); j++){ |
|  | printf("A[%d][%d] " ,i,j); |
|  | scanf("%f",&A[i][j]);}} |
|  | for(j=1;j<=n;j++){ |
|  | for(i=1;i<=n;i++){ |
|  | if(i>j){ |
|  | c=A[i][j]/A[j][j]; |
|  | for(k=1;k<=n+1 ;k++){ |
|  | A[i][k]=A[i][k]-c\*A[j][k]; |
|  | }}}} |
|  | x[n]=A[n][n+1]/A[n][n]; |
|  |  |
|  |  |
|  | for(i=n-1; i>=1; i--){ |
|  | sum=0; |
|  | for(j=i+1 ;j<=n;j++){ |
|  | sum =sum+A[i][j]\*x[j]; |
|  | x[i]=(A[i][n+1]-sum)/A[i][i];}} |
|  | printf("InThe solution is: \n"); |
|  | for(i=1; i<=n ; i++) |
|  | printf ( "\nx%d=%f\t",i,x[i]); |
|  | return 0;} |