Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»
Факультет программной инженерии и компьютерной техники

Лабораторная работа 1 по дисциплине

«Вычислительная математика»

Вариант № 10

Выполнил:
Мамонтов Г. А.

Преподаватели:
Машина Е. А.
Малышева Т. А.

Санкт-Петербург, 2025 г

Цель работы

Целью данной работы является изучение численных методов решения нелинейных уравнений и их систем, найти корни заданного нелинейного уравнения/системы нелинейных уравнений, выполнить программную реализацию методов.

Описание метода, расчетные формулы

В моем варианте рассматривается метод Гаусса-Зейделя, приведем его описание и формулы, использующиеся при выполнении этого метода.

Метод Гаусса-Зейделя является модификацией метода простой итерации и обеспечивает более быструю сходимость к решению систем уравнений. Так же как и в методе простых итераций строится эквивалентная СЛАУ и за начальное приближение принимается вектор правых частей (как правило, но может быть выбран и нулевой, и единичный вектор): $x_i^0 = (d_1, d_2, \dots, d_n)$.

$$
\begin{gathered}
x_1 = c_{11}x_1 + c_{12}x_2 + \dots + c_{1n}x_n + d_1 \\
x_2 = c_{21}x_1 + c_{22}x_2 + \dots + c_{2n}x_n + d_2 \\
\dots \\
\dots \\
x_n = c_{n1}x_1 + c_{n2}x_2 + \dots + c_{nn}x_n + d_n
\end{gathered}
$$

Идея метода: при вычислении компонента $x_i^{(k+1)}$ на (k+1)-й итерации используются $x_1^{(k+1)}$, $x_2^{(k+1)}$, … , $x_{i-1}^{(k+1)}$, уже вычисленные на (k+1)-й итерации.

Значения остальных компонент $x_{i+1}^{(k+1)}$, $x_{i+2}^{(k+1)}$, … , $x_n^{(k+1)}$ берутся из предыдущей итерации.

Тогда приближения к решению системы методом Зейделя определяются следующей системой равенств:

$$x_1^{(k+1)} = c_{11}x_1^{(k)} + c_{12}x_2^{(k)} + \dots + c_{1n}x_n^{(k)} + d_1$$

$$x_2^{(k+1)} = c_{21}x_1^{(k+1)} + c_{22}x_2^{(k)} + \dots + c_{2n}x_n^{(k)} + d_2$$

$$x_3^{(k+1)} = c_{31}x_1^{(k+1)} + c_{32}x_2^{(k+1)} + c_{33}x_3^{(k)} \dots + c_{3n}x_n^{(k)} + d_3$$

…………………………………………………………………

$$x_n^{(k+1)} = c_{n1}x_1^{(k+1)} + c_{n2}x_2^{(k+1)} + \dots + c_{n\,n-1}x_{n-1}^{(k+1)} + c_{n\,n}x_n^{(k)} + d_n$$

Рабочая формула метода Гаусса-Зейделя:

$$x_j^{(k+1)} = \frac{b_i}{a_{ii}} - \sum_{j=1}^{i-1}\frac{a_{ij}}{a_{ii}}x_j^{k+1} - \sum_{j=i+1}^{n}\frac{a_{ij}}{a_{ii}}x_j^{k} \quad i = 1, 2, \dots, n$$

Листинг программы

```
private static double[] gaussSeidel(double[][] A, double[] b, double tol, List<Double> errors) {
    int n = A.length;
    double[] x = new double[n]; // Начальное приближение (нулевое)
    int iterations = 0;
    double maxError;
    do {
        maxError = 0;
        for (int i = 0; i < n; i++) {
            double sum = 0;
            for (int j = 0; j < n; j++) {
                if (i != j) {
                    sum += A[i][j] * x[j]; // Используем уже обновленные значения
```

```
                }
            }
            double newX = (b[i] - sum) / A[i][i];
            maxError = Math.max(maxError, Math.abs(newX - x[i])); // Считаем максимальную ошибку
            x[i] = newX; // Обновляем x[i]
        }
        errors.add(maxError);
        iterations++;
    } while (maxError > tol);
    System.out.println("Итерации: " + iterations);
    return x;
}
```

Примеры и результаты работы программы

Пример работы программы при введении матрицы с диагональным преобладанием:

```
Введите размерность матрицы (n <= 20): 3
Прочитать данные из файла (f) или с клавиатуры (k)? k
Введите коэффиценты матрицы ряд за рядом:
4 -1 0
-1 4 -1
0 -1 4
Введите вектор b:
15 10 10
Введите точность: 0,001
Норма матрицы 6.0
Итерации: 6
Решение: [4.910640716552734, 4.642820358276367, 3.660705089569092]
Вектор погрешностей: [3.75, 1.0546875, 0.263671875, 0.032958984375, 0.004119873046875, 5.14984130859375E-4]
```

Пример работы программы при введении матрицы без диагонального преобладания:

```
Введите размерность матрицы (n <= 20): 3
Прочитать данные из файла (f) или с клавиатуры (k)? k
Введите коэффиценты матрицы ряд за рядом:
1 1 1
1 1 1
1 1 1
Введите вектор b:
1 1 1
Введите точность: 0,001
В матрице нет диагонального преобладания. Дальнейшее выполнение программы невозможно.
```

Выводы

Программа успешно выполняет метод Гаусса - Зейделя для поиска решения системы линейных уравнений. Во время работы над выполнением поставленной задачи мною был понят алгоритм программирования математических операций, заточенных на неоднократном повторении одной ключевой формулы и был получен опыт программирования численных методов.