

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ Н. Э. БАУМАНА
Факультет информатики и систем управления
Кафедра теоретической информатики и компьютерных технологий

Лабораторная работа №6
по курсу «Численные методы»

«Решение неоднородного дифференциального
уравнения второго порядка
в постановке краевой задачи»

Выполнил:
студент группы ИУ9-62
Беляев А. В.

Проверила:
Домрачева А. Б.

Москва 2016

Содержание

1	Постановка задачи	3
2	Необходимые теоретические сведения	3
3	Текст программы	4
4	Результаты	5
5	Выводы	5

1 Постановка задачи

Необходимо решить дифференциальное уравнение, используя метод прогонки (для трехдиагональной матрицы). Пусть дифференциальное уравнение при этом задано в постановке краевой задачи:

$$\begin{cases} y''(x) + p(x)y'(x) + q(x)y(x) = f(x), x \in [a, b] \\ y(a) = A \\ y(b) = B \end{cases}$$

2 Необходимые теоретические сведения

Для начала рассмотрим формулу Тейлора:

$$y(x) = y(a) + \frac{y'(a)}{1!}(x-a) + \frac{y''(a)}{2!}(x-a)^2 + \frac{y'''(a)}{3!}(x-a)^3 + \dots + O((x-a)^4) \quad (1)$$

Введем следующие обозначения:

$$h = \frac{b-a}{n}, f_i = f(x_i), p_i = p(x_i), q_i = q(x_i)$$

где n - число разбиений отрезка $[a, b]$, а $x_i = a + i * h$.

С учетом введенных выше обозначений формула Тейлора принимает следующий вид:

$$y_{i-1} = y_i - y'_i h + \frac{y''_i h^2}{2} - \frac{y'''_i h^3}{6} + O(h^4) \quad (2)$$

$$y_{i+1} = y_i + y'_i h + \frac{y''_i h^2}{2} + \frac{y'''_i h^3}{6} + O(h^4) \quad (3)$$

Сложив (2) и (3), найдем выражение для y''_i . Вычтя (3) из (2), найдем y'_i :

$$y''_i \approx \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$$

$$y'_i \approx \frac{y_{i+1} - y_{i-1}}{2h}$$

Подставив полученные для y''_i и y'_i значения в уравнение из системы (1), получаем новую систему (4).

$$\begin{cases} (q_1 h^2 - 2)y_1 + (1 + \frac{p_1 h}{2})y_2 = f_1 h^2 - A(1 - \frac{p_1 h}{2}) \\ \dots \\ (1 - \frac{p_i h}{2})y_{i-1} + (q_i h^2 - 2)y_i + (1 + \frac{p_i h}{2})y_{i+1} = f_i h^2 \\ \dots \\ (1 - \frac{p_{n-1} h}{2})y_{n-1} + (q_{n-1} h^2 - 2)y_n = f_{n-1} h^2 - B(1 + \frac{p_{n-1} h}{2}) \end{cases} \quad (4)$$

Решив систему методом прогонки, получим значения функции y в $x_i \in [a, b]$.

3 Текст программы

Для написания программы был использован язык Javascript.

```
var y = x => 3*x*x*x;
var dy = x => 9*x*x;
var ddy = x => 18*x;

var pFunc = x => 1;
var qFunc = x => -1;

N = 11;

x_A = 1.0;
x_B = 2.0;

y_A = y(x_A);
y_B = y(x_B);

h = (x_B - x_A) / (N - 1);

f = [], q = [], p = []; // size = n

function solveMatrix(n, a, b, c, d) {
    var alpha = [], beta = [], res = []; // n

    alpha[0] = -c[0] / b[0];
    beta[0] = d[0] / b[0];

    for (var i = 1; i < n; i++) {
        alpha[i] = (-c[i] / (a[i] * alpha[i-1] + b[i]));
        beta[i] = (d[i] - a[i] * beta[i-1]) / (a[i] * alpha[i-1] + b[i]);
    }

    for (var i = n-2; i >= 0; i--) res[i] = alpha[i] * res[i+1] + beta[i];

    return res;
}

function countFunc(n) {
    for (var i = 0; i < n; i++) {
        var x = x_A + i*h;

        f[i] = ddy(x) + pFunc(x)*dy(x) + qFunc(x)*y(x);
        p[i] = pFunc(x);
        q[i] = qFunc(x);
    }
}

function solveEquation(n) {
    var a = [], b = [], c = [], d = []; // n-2
    var e = [], res = []; // n

    for (var i = 1; i < n-3; i++) {
        a[i] = 1 - p[i+1]*h/2;
        b[i] = q[i+1]*h*h - 2;
        c[i] = 1 + p[i+1]*h/2;
        d[i] = f[i+1]*h*h;
    }

    a[0] = 0;
    b[0] = q[1]*h*h - 2;
    c[0] = 1 + p[1]*h/2;
    d[0] = f[1]*h*h - y_A*(1 - p[1]*h/2);

    a[n-3] = 1 - p[n-2]*h/2;
    b[n-3] = q[n-2]*h*h - 2;
    c[n-3] = 0;
    d[n-3] = f[n-2]*h*h - y_B*(1 + p[n-2]*h/2);

    e = solveMatrix(n-2, a, b, c, d);

    for (var i = 0; i < n-2; i++) res[i+1] = e[i];

    res[0] = y_A;
    res[n-1] = y_B;

    for (var i = 0; i < n; i++)
        result += "[" + i + "]: x: " + (x_A + i*h) +
            " y: " + y(x_A + i*h) +
            " res: " + res[i] +
            " delta: " + Math.abs(y(x_A + i*h) - res[i]) + "\n";

    console.log(result);
}

var lab6Main = () => { countFunc(N); solveEquation(N) };
```

4 Результаты

В результате работы программы были получены следующие результаты:

Для функции $y(x) = 3x^3$ и входных параметров

$$p(x) = 1, q(x) = -1$$

$$f(x) = -3x^3 + 9x^2 + 18x$$

на отрезке $[1; 2]$ с количеством разбиений $= 10$

были получены следующие результаты:

Таблица 1: Результаты работы 1

x	y ожидаемое	y полученное	Абс. погр
1.0	3.000000	3.000000	0.000000
1.1	3.993000	3.994337	0.001337
1.2	5.184000	5.186371	0.002371
1.3	6.591000	6.593998	0.002998
1.4	8.232000	8.235308	0.003308
1.5	10.125000	10.128334	0.003334
1.6	12.288000	12.291103	0.003103
1.7	14.739000	14.741639	0.002639
1.8	17.496000	17.497958	0.001958
1.9	20.577000	20.578075	0.001075
2.0	24.000000	24.000000	0.000000

5 Выводы

В ходе работы был реализован метод решения дифференциального уравнения с использованием метода прогонки.

Как видно из результатов тестирования, данный метод вычисляет результат с достаточно высокой точностью. Погрешность обусловлена количеством членов ряда Тейлора, в которые раскладывается функция, и может быть уменьшена путем разбиения отрезка на большее число точек.