Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления»

КАФЕДРА «Теоретическая информатика и компьютерные технологии»

ОТЧЕТ

по лабораторной работе № 15

по курсу «Численные методы»

на тему: «Решение краевой задачи методом стрельбы»

Вариант № 4

Студент ИУ9-61Б Афанасьев И.
(Группа) (Подпись, дата) (И. О. Фамилия)

Преподаватель Домрачева А. Б.
(Подпись, дата) (И. О. Фамилия)

2024 г.

1 Постановка задачи

1. Решить аналитически задачу Коши

$$y'' + py' + qy = f(x), \quad y(0) = y_0, \quad y'(0) = y'_0$$

и по найденному решению задачи Коши $y(x)$ вычислить $b = y(1)$.

2. Используя метод стрельбы найти численное решение (x_i, y_i), $i = 0, \ldots, n$, краевой задачи для того же уравнения с краевыми условиями $y(0) = a$, $y(1) = b$ при $n = 10$.

3. Найти погрешность численного решения $||y - \tilde{y}|| = \max |y(x_i) - \tilde{y}_i|$, $0 \leq i \leq n$.

Индивидуальный вариант: $p = -1$, $q = 0$, $f(x) = 2(1 - x)$, $y_0 = 1$, $y'_0 = 1$.

2 Основные теоретические сведения

Рассмотрим краевую задачу для линейного дифференциального уравнения второго порядка:

$$y'' + p(x)y' + q(x)y = f(x), \quad y(a) = A, \quad y(b) = B.$$

Общее решение этого уравнения имеет вид

$$y_{cs} = C_1 y_1 + C_2 y_2 + y_{ps},$$

где y_{cs} — общее решение неоднородного уравнения; $C_1 y_1 + C_2 y_2$ — общее решение соответствующего однородного уравнения; y_1 и y_2 — линейно независимые частные решения однородного уравнения; C_1 и C_2 — произвольные постоянные; y_{ps} — частное решение неоднородного уравнения. Величины C_1 и C_2 определяют из системы уравнений

$$C_1 y_1(a) + C_2 y_2(a) + y_{ps}(a) = A,$$

$$C_1 y_1(b) + C_2 y_2(b) + y_{ps}(b) = B.$$

Если частное решение неоднородного уравнения удовлетворяет условию $y_{ps}(a) = A$, а одно из частных решений однородного уравнения, например, $y_1(x)$, условию $y_1(a) = 0$, то первое уравнение системы принимает вид

$$C_1 \cdot 0 + C_2 y_2(a) + A = A,$$

и следовательно, $C_2 = 0$. Постоянную C_1 определяют из второго уравнения $C_1 y_1(b) + y_{ps}(b) = B$. Описанный метод называют *методом стрельбы.*

Рассмотрим его сеточный аналог, реализуемый в данной работе. Для этого разобьём отрезок $[a, b]$ на n частей точками $x_0, x_1, \ldots, x_n$, где $x_i = a + ih$, $h = \frac{b-a}{n}$, а производные исходного дифференциального уравнения во всех внутренних точках заменим их разностными аналогами:

$$y'_i = \frac{y_{i+1} - y_{i-1}}{2h}, \quad y''_i = \frac{y_{i+1} - 2y_i + y_{i+1}}{h^2}, \quad i = 1, 2, \ldots, n-1.$$

Будем искать решения, удовлетворяющие условиям

$$y_0[0] = A, \quad y_0[1] = D_0, \quad y_1[0] = 0, \quad y_1[1] = D_1 \neq 0$$

(используется обозначение $y_i[j] = y_i(x_j)$). Для уменьшения вычислительной погрешности обычно берут $D_0 = A + \mathcal{O}(h)$, $D_1 = \mathcal{O}(h)$.

Для определения y_0 и y_1 получим уравнения

$$\frac{y_0[i+1] - 2y_0[i] + y_0[i-1]}{h^2} + p_i \frac{y_0[i+1] - y_0[i-1]}{2h} + q_i y_0[i] = f_i,$$

$$\frac{y_1[i+1] - 2y_1[i] + y_1[i-1]}{h^2} + p_i \frac{y_1[i+1] - y_1[i-1]}{2h} + q_i y_1[i] = 0.$$

Отсюда

$$y_0[i+1] = \frac{f_i h^2 + (2 - q_i h^2) y_0[i] - (1 - \frac{p_i h}{2}) y_0[i-1]}{1 + p_i \frac{h}{2}},$$

$$y_1[i+1] = \frac{(2 - q_i h^2) y_1[i] - (1 - \frac{p_i h}{2}) y_1[i-1]}{1 + \frac{p_i h}{2}},$$

$$i = 1, 2, \ldots, n-1.$$

Затем находим C_1 из уравнения $y_n[n] + C_1 y_1[n] = B$, т.е. $C_1 = \frac{B - y_0[n]}{y_1[n]}$. Искомое решение задачи находим теперь по формулам

$$y[t] = y_0[i] + C_1 y_1[i], \quad i = 0, 1, \ldots, n.$$

3 Реализация

В листинге 3.1 представлен исходный код программы на языке C++.

Листинг 3.1 – Исходный код программы на языке C++

```cpp
#include <cassert>
#include <iomanip>
#include <cmath>
#include <ios>
#include <iostream>
#include <numbers>
#include <vector>

int main() {
  const auto y = [](const double x) -> double { return
     std::exp(x) + x * x; };
  constexpr auto p = [](const double x) -> double { return -1; };
  constexpr auto q = [](const double x) -> double { return 0; };
  constexpr auto f = [](const double x) -> double { return 2 *
     (1 - x); };

  constexpr double a = 0, b = 1;
  constexpr double A = 1, B = std::numbers::e + 1;
  constexpr std::size_t n = 10;

  constexpr auto h = (b - a) / n;
  constexpr auto h_sqr = h * h;
  constexpr auto h_half = h / 2;

  std::vector<double> xs;
  xs.reserve(n + 1);

  auto x = a;
  for (std::size_t i = 0; i <= n; ++i) {
    xs.push_back(x);
    x += h;
  }

  constexpr auto O_h = 1e-3;
  constexpr auto D0 = A + O_h;
  constexpr auto D1 = O_h;

```

```
std::vector<double> y0s, y1s;
y0s.reserve(n + 1);
y1s.reserve(n + 1);

y0s.push_back(A);
y0s.push_back(D0);

y1s.push_back(0);
y1s.push_back(D1);

for (std::size_t i = 1; i < n; ++i) {
  const auto pi = p(xs[i]), qi = q(xs[i]), fi = f(xs[i]);
  y0s.push_back((fi * h_sqr + (2 - qi * h_sqr) * y0s[i] -
                 (1 - pi * h_half) * y0s[i - 1]) /
                (1 + pi * h_half));
  y1s.push_back(((2 - qi * h_sqr) * y1s[i] - (1 - pi * h_half)
     * y1s[i - 1]) /
                (1 + pi * h_half));
}

const auto C1 = (B - y0s[n]) / y1s[n];

std::vector<double> ys;
ys.reserve(n + 1);

for (std::size_t i = 0; i <= n; ++i) {
  ys.push_back(y0s[i] + C1 * y1s[i]);
}

const auto w = 9;

std::cout << std::setw(w) << "i";
for (std::size_t i = 0; i <= n; ++i) {
  std::cout << std::setw(w) << i;
}
std::cout << std::endl;

std::cout << std::setw(w) << "x";
for (auto&& x : xs) {
  std::cout << std::setw(w) << x;
}
```

```
  std::cout << std::endl;

  std::cout << std::setw(w) << "y_an";
  for (std::size_t i = 0; i <= n; ++i) {
    std::cout << std::setw(w) << std::fixed << y(xs[i]);
  }
  std::cout << std::endl;

  std::cout << std::setw(w) << "y_nm";
  for (auto&& y : ys) {
    std::cout << std::setw(w) << std::fixed << y;
  }
  std::cout << std::endl;

  std::cout << std::setw(w) << "error";
  for (std::size_t i = 0; i <= n; ++i) {
    std::cout << std::setw(w) << std::fixed << std::abs(y(xs[i]) - ys[i]);
  }
  std::cout << std::endl;
}
```

4 Результаты

В листинге 4.1 представлены результаты работы программы.

Листинг 4.1 – Результаты работы программы

```
       i        0        1        2        3        4        5
       x        0      0.1      0.2      0.3      0.4      0.5
    y_an 1.000000 1.115171 1.261403 1.439859 1.651825 1.898721
    y_nm 1.000000 1.115124 1.261314 1.439735 1.651673 1.898553
   error 0.000000 0.000047 0.000088 0.000124 0.000151 0.000168

       6        7        8        9       10
     0.6      0.7      0.8      0.9        1
2.182119 2.503753 2.865541 3.269603 3.718282
2.181946 2.503591 2.865409 3.269523 3.718282
0.000173 0.000162 0.000132 0.000080 0.000000
```