



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 и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Летучка №1
по курсу «Численные методы линейной алгебры»
«Метод прогонки для трёхдиагональной матрицы.»

Студент группы ИУ9-71Б Окутин Д.А.

Преподаватель Посевин Д. П.

Москва 2024

1 Задание

Реализовать метод прогонки для трёхдиагональной матрицы, вычислить погрешность по Евклидовой норме.

2 Реализация

Исходный код программы представлен в листингах 1– ??.

Листинг 1 — Вспомогательные функции

```
1 using Random
2
3 function euclidean_norm(vec::Vector)
4     return sqrt(sum(vec.^2))
5 end
6
7 function generate_vector(l::Int, r::Int, n::Int)
8     return rand(n) .* (r - l) .+ l
9 end
10
11 function generate_rhs_vector(A, B, C, x)
12     n = length(x)
13     d = zeros(n)
14     d[1] = B[1] * x[1] + C[1] * x[2]
15     for i in 2:n - 1
16         d[i] = A[i - 1] * x[i - 1] + B[i] * x[i] + C[i] * x[i + 1]
17     end
18     d[n] = A[n - 1] * x[n - 1] + B[n] * x[n]
19
20     return d
21 end
```

Листинг 2 — Метод прогонки

```
1
2 function forward(A, B, C, d)
3     n = length(d)
4     alphas = zeros(n)
5     betas = zeros(n)
6     alphas[1] = -C[1] / B[1]
7     betas[1] = d[1] / B[1]
8     for i in 2:n
9         if i == n
10             alphas[i] = 0.0
11         else
12             alphas[i] = -C[i] / (alphas[i - 1] * A[i - 1] + B[i])
13         end
14         betas[i] = (d[i] - A[i - 1] * betas[i - 1]) / (alphas[i - 1] * A
15 [i - 1] + B[i])
16     end
17     return alphas, betas
18 end
19
20
21 function backward(alphas, betas)
22     n = length(betas)
23     x = zeros(n)
24     x[n] = betas[n]
25     for i in n-1:-1:1
26         x[i] = alphas[i] * x[i + 1] + betas[i]
27     end
28
29     return x
30 end
31
32 function progonka(a, b, c, d)
33     alphas, betas = forward(a,b,c,d)
34     return backward(alphas, betas)
35 end
```

Листинг 3 — Тест

```
1
2 n = 10
3
4 a = generate_vector(-10,10,n-1)
5 b = generate_vector(-10,10,n)
6 c = generate_vector(-10,10,n-1)
7 x = ones(n)
8 d = generate_rhs_vector(a,b,c,x)
9
10 res = progonka(a,b,c,d)
11
12 println("Result: $res")
13
14 shift = euclidean_norm(x - res)
15
16 println("Razn: $shift")
```

3 Результаты

Результат запуска методов представлены на рисунке 1.

Результат метода прогонки: [0.999999999999984, 0.999999999999956, 1.0000000000000013, 1.0000000000000007, 0.999999999999997, 1.0000000000000004, 0.999999999999998, 0.999999999999997, 0.999999999999994, 0.999999999999986]
Погрешность: 1.6744872289733755e-14

Рис. 1 — Результат работы

4 Выводы

В результате выполнения данной работы был реализован метод прогонки, а также была вычислена погрешность вычислений, которая, как и ожидалось, оказалось не нулевая.