

```

1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5
6 vector<vector<float>> sumarMatrices(vector<vector<float>> matriz1, vector<vector<float>> matriz2) {
7     int n = matriz1.size();
8
9     vector<vector<float>> resultado(n, vector<float>(n, 0));
10
11     for (int i = 0; i < n; i++) {
12         for (int j = 0; j < n; j++) {
13             resultado[i][j] = matriz1[i][j] + matriz2[i][j];
14         }
15     }
16
17     return resultado;
18 }
19
20 vector<vector<float>> multiplicarMatrices(vector<vector<float>> matriz1, vector<vector<float>> matriz2) {
21     int n = matriz1.size();
22
23     vector<vector<float>> resultado(n, vector<float>(n, 0));
24
25     for (int i = 0; i < n; i++) {
26         for (int j = 0; j < n; j++) {
27             for (int k = 0; k < n; k++) {
28                 resultado[i][j] += matriz1[i][k] * matriz2[k][j];
29             }
30         }
31     }
32
33     return resultado;
34 }
35
36 int calcularTraza(vector<vector<float>> matriz) {
37     int traza = 0;
38     int n = matriz.size();
39
40     for (int i = 0; i < n; i++) {
41         traza += matriz[i][i];
42     }
43
44     return traza;
45 }
46
47 vector<vector<float>> matrizIdentidad(int n) {
48     vector<vector<float>> identidad(n, vector<float>(n, 0));
49     for (int i = 0; i < n; i++) {
50         identidad[i][i] = 1;
51     }
52     return identidad;
53 }
54
55 vector<vector<float>> multiplicarEscalar(float l, vector<vector<float>> matriz) {
56     int n = matriz.size();
57
58     vector<vector<float>> resultado(n, vector<float>(n, 0));
59     for (int i = 0; i < n; i++) {
60         for (int j = 0; j < n; j++) {
61             resultado[i][j] = matriz[i][j] * l;
62         }
63     }
64
65     return resultado;
66 }
67
68 void printMatriz(vector<vector<float>> matrix)
69 {
70     for (int i = 0; i < matrix.size(); i++)
71     {
72         for (int j = 0; j < matrix[i].size(); j++)
73             cout << matrix[i][j] << " ";
74         cout << endl;
75     }
76 }
77
78 vector<float> faddeev(vector<vector<float>> A, vector<vector<float>> B, int n)
79 {
80     float bi = 1;
81     vector<float> bi_vector;
82     bi_vector.push_back(bi);
83     //cout<<"b"<<n<<" = "<<bi<<endl;
84
85     for(float k=1; k<=n; k++)
86     {
87         g = sumarMatrices(multiplicarMatrices(A,B), multiplicarEscalar(bi, matrizIdentidad(n)));
88         bi = -1/k * calcularTraza(multiplicarMatrices(A,B));
89         //cout<<"b"<<n-k<<" = "<<bi<<endl;
90         bi_vector.push_back(bi);
91     }
92
93     return bi_vector;
94 }
95
96 int main()
97 {
98     vector<vector<float>> A =
99     {
100         {
101             {3, 1, 5},
102             {3, 3, 1},
103             {4, 6, 4}};
104
105     /*
106     vector<vector<float>> A =
107     {
108         {
109             {3, 2, 4},
110             {2, 0, 2},
111             {4, 2, 3}};
112
113     */
114
115     int n = A.size();
116     vector<vector<float>> B(n, vector<float>(n, 0));
117
118     vector<float> bi_vector = faddeev(A, B, n);
119
120     for(int i=0; i<=n; i++)
121     {
122         cout<<"b"<<n-i<<" = "<<bi_vector[i]<<endl;
123     }
124 }
125

```

RunnerFile }

b3 = 1

b2 = -10

b1 = 4

b0 = -40

PS C:\Users\luisa\OneDrive - up.edu.mx\Docu