

```

1 // Autor: Rannier Salles de Almeida
2 // Prova PCC104 - 01
3 // Algoritmo: Busca em Largura e Profundidade
4 #include <iostream>
5 #include <vector>
6 #include <stack>
7 #include <queue>
8
9 using namespace std;
10
11 vector<vector<int>> InserirMatriz(int No) {
12     vector<vector<int>> AdjMatriz(No, vector<int>(No));
13
14     for (int i = 0; i < No; i++) {
15         for (int j = 0; j < No; j++) {
16             cout << "A" << i << j << ": ";
17             cin >> AdjMatriz[i][j];
18         }
19     }
20
21     return AdjMatriz;
22 }
23
24 void ImprimeMatriz(vector<vector<int>> AdjMatriz, int No) {
25     cout << endl << "Matriz de Adjacencia:" << endl;
26
27     for (int i = 0; i < No; i++) {
28         for (int j = 0; j < No; j++) {
29             if (j != No - 1)
30                 cout << AdjMatriz[i][j] << "\t";
31             else
32                 cout << AdjMatriz[i][j] << endl;
33         }
34     }
35 }
36
37 void MostrarPilha(stack<int> pilha) {
38     cout << endl;
39     cout << "Elementos da Pilha:";
40     while (!pilha.empty()) {
41         cout << pilha.top() << " ";
42         pilha.pop();
43     }
44 }
45
46 void MostrarFila(queue<int> fila) {
47     cout << endl;
48     cout << "Elementos da Fila:";
49     while (!fila.empty()) {
50         cout << fila.front() << " ";
51         fila.pop();
52     }
53 }
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84

```

```

85 void BuscaProfundidade(vector<vector<int>>& AdjMatriz, int No, int Inicial[2], int alvo) {
86     vector<bool> visitados(No, false);
87     stack<int> pilha;
88     int inicio = Inicial[0];
89
90     cout << "Busca em Profundidade a partir do No " << inicio << " para o alvo " << alvo << ": ";
91
92     pilha.push(inicio);
93
94
95     while (!pilha.empty()) {
96         int NoAtual = pilha.top();
97         cout << "No atual: " << NoAtual;
98
99         pilha.pop();
100
101
102         if (!visitados[NoAtual]) {
103             cout << NoAtual << " ";
104             visitados[NoAtual] = true;
105
106             if (NoAtual == alvo) {
107                 cout << "Alvo encontrado!" << endl;
108                 cout << endl;
109                 cout << endl;
110                 return;
111             }
112
113             for (int i = 0; i < No; i++) {
114                 if (AdjMatriz[NoAtual][i] && !visitados[i] && NoAtual != i) {
115                     pilha.push(i);
116                 }
117             }
118         }
119     }
120
121     cout << "Alvo não encontrado." << endl;
122     cout << endl;
123     cout << endl;
124 }
125
126 void Primeiro_No(const vector<vector<int>>& AdjMatriz, int No, int inicial[2]) {
127     for (int i = 0; i < No; i++) {
128         for (int j = 0; j < No; j++) {
129             if (AdjMatriz[i][j] != 0) {
130                 cout << endl << "No Inicial: A" << i << j << endl;
131                 inicial[0] = i;
132                 inicial[1] = j;
133                 return;
134             }
135         }
136     }
137 }
138
139 void BuscaLargura(vector<vector<int>> AdjMatriz, int No, int Inicial[2], int alvo) {
140     queue<int> q;
141     vector<bool> visitados(No, false);
142
143     int inicio = Inicial[0]; // O primeiro elemento de Inicial é o vértice de início
144     cout << "Busca em Largura a partir do No " << inicio << " para o alvo " << alvo << ": ";
145
146     q.push(inicio);
147     visitados[inicio] = true;
148
149     while (!q.empty()) {
150         int NoAtual = q.front();
151         q.pop();
152
153
154         cout << NoAtual << " ";
155
156         if (NoAtual == alvo) {
157             cout << "Alvo encontrado!" << endl;
158             cout << endl;
159             cout << endl;
160             return;
161         }
162
163         for (int i = 0; i < No; i++) {
164             if (AdjMatriz[NoAtual][i] && !visitados[i]) {
165                 q.push(i);
166                 visitados[i] = true;
167             }
168         }

```

```

169     }
170
171     cout << "Alvo não encontrado." << endl;
172     cout << endl;
173     cout << endl;
174 }
175
176 int main() {
177     int No;
178     int Inicial[2];
179     int alvo;
180
181     cout << "Entre com a quantidade de nos: ";
182     cin >> No;
183
184     cout << "Insira os valores da matriz Aij" << endl;
185     vector<vector<int>> AdjMatriz = InserirMatriz(No);
186     ImprimeMatriz(AdjMatriz, No);
187
188     Primeiro_No(AdjMatriz, No, Inicial);
189
190     cout << "Entre com o no alvo: ";
191     cin >> alvo;
192
193     BuscaLargura(AdjMatriz, No, Inicial, alvo);
194     BuscaProfundidade(AdjMatriz, No, Inicial, alvo);
195
196     return 0;
197 }
198

```