

PRÁTICA 4 – Trabalho 2 Exercício 4: Busca Linear em Vetores

Nome: Davi Gabriel Domingues

Número USP: 15447497

O objetivo desta atividade é o de ajudar uma prefeitura de uma cidade hipotética a fazer uma pesquisa sobre os domicílios na área urbana. Em cada domicílio, são coletados dados sobre renda média mensal, número total de ocupantes, endereço e número de ocupantes em idade escolar.

Para tanto, foi – se solicitado um programa, em C++, que contivesse um menu que permitisse:

- (1) A Digitação dos dados de um domicílio;
- (2) A Busca dos dados do domicílio quando um determinado endereço é digitado;
- (3) A Saída do sistema de seleção de comandos.

Como condições para a execução do programa citado, estas observações foram solicitadas:

- a) Uso de estrutura/"struct" (para os domicílios).
- b) Os dados devem estar em um vetor (de domicílios).
- c) Deve-se utilizar busca linear com sentinela no item 2.

Dessa forma, o programa a seguir foi desenvolvido, a fim de ser tomado como resposta para a situação-problema proposta na Prática:

```
1  #include <iostream>
2  #include <limits>
3  using namespace std;
4
5  struct Domicilio{
6      float rendaMensal;
7      int totalOcupantes;
8      string endereco;
9      int ocupantesIdadeEscolar;
10 };
11
12 bool preencherInformacaoDomicilio(Domicilio& domicilio) {
13     cout << "Renda mensal do domicílio: ";
14     cin >> domicilio.rendaMensal;
15     if (domicilio.rendaMensal <= 0) {
16         cout << "Dado invalido! Insercao interrompida..." << endl;
17         cin.ignore(numeric_limits<streamsize>::max(), '\n'); // Ignora a linha
18         return false;
19     }
20     cin.ignore(numeric_limits<streamsize>::max(), '\n');
21 }
```

```

22     cout << "Total de ocupantes do domicilio: ";
23     cin >> domicilio.totalOcupantes;
24     if (domicilio.totalOcupantes <= 0) {
25         cout << "Dado invalido! Insercao interrompida..." << endl;
26         cin.ignore(numeric_limits<streamsize>::max(), '\n');
27         return false;
28     }
29     cin.ignore(numeric_limits<streamsize>::max(), '\n');
30
31     cout << "Endereco do domicilio: ";
32     getline(cin, domicilio.endereco);
33     if (domicilio.endereco.empty()) {
34         cout << "Dado invalido! Insercao interrompida..." << endl;
35         return false;
36     }
37
38     cout << "Ocupantes em idade escolar do domicilio: ";
39     cin >> domicilio.ocupantesIdadeEscolar;
40     if (domicilio.ocupantesIdadeEscolar <= 0 || domicilio.ocupantesIdadeEscolar >= domicilio.totalOcupantes) {
41         cout << "Dado invalido! Insercao interrompida..." << endl;
42         cin.ignore(numeric_limits<streamsize>::max(), '\n');
43         return false;
44     }
45     cin.ignore(numeric_limits<streamsize>::max(), '\n');
46
47     return true;
48 }
49
50 void encontrarDadosDomicilio(Domicilio domicilio[], int i, string& endereco_desejado){
51     domicilio[i].endereco = endereco_desejado; // atribuicao ao sentinela da busca linear.
52     for (int j = 0; j < i; j++){
53         if(domicilio[j].endereco == domicilio[i].endereco){
54             cout<<"Endereco localizado, dados encontrados!"<<endl;
55             cout<<"Renda mensal: "<<domicilio[j].rendaMensal<<endl;
56             cout<<"Total de ocupantes: "<<domicilio[j].totalOcupantes<<endl;
57             cout<<"Ocupantes em idade escolar: "<<domicilio[j].ocupantesIdadeEscolar<<endl;
58             return;
59         }
60     }
61     cout<<"Endereco nao localizado, dados inexistentes!";
62 }

```

```

63
64 int main(){
65     int opcao = 0, i = 0, total_domicilios;
66     string endereco_desejado;
67
68     cout<<"Informe o total de domicilios: ";
69     cin>>total_domicilios;
70     cout<<endl;
71
72     Domicilio* domicilio = new Domicilio[total_domicilios + 1]; //vetor que comporta o sentinela
73
74     do{
75         cout<<"Informe o que deseja fazer:"<<endl;
76         cout<<"(1) Digitar os dados de um domicilio"<<endl;
77         cout<<"(2) Buscar os dados de um domicilio, a partir de um endereco"<<endl;
78         cout<<"(3) Sair"<<endl;
79         cin>>opcao;
80
81         switch(opcao){
82             case 1:
83                 if (i < total_domicilios){

```

```

84         if (i < total_domicilios){
85             if (preencherInformacaoDomicilio(domicilio[i])) i++;
86         }
87
88         else cout<<"Numero maximo de domicilios alcancado, impossivel adicionar informacoes!"<<endl;
89         break;
90
91         case 2:
92             cout<<"Informe o endereco desejado para saber os dados associados: ";
93             getline(cin, endereco_desejado);
94
95             encontrarDadosDomicilio(domicilio, i, endereco_desejado);
96             break;
97
98         case 3:
99             cout<<endl<<"Saindo..."<<endl;
100            break;
101
102            default:
103                cout<<"Opcao invalida, tente novamente..."<<endl;
104            }
105
106            cin.ignore(numeric_limits<streamsize>::max(), '\n'); // limpa o buffer antes de um novo loop
107        } while(opcao != 3);
108
109        delete[] domicilio;
110
111        return 0;
112    }

```