



Pontifícia Universidade Católica de Minas Gerais  
Curso de Ciência da Computação  
Disciplina: Algoritmos e Estruturas de Dados I  
Prof. Felipe Cunha

# Laboratório 11

---

main.cpp

```
1 #include <iostream>
2 #include "elevador.h"
3
4 void menu(){
5     system("clear");
6     cout << "*****" << endl;
7     cout << "          MENU DE OPCOES          " << endl;
8     cout << "*****" << endl;
9     cout << "1) para o Elevador subir" << endl;
10    cout << "2) para o Elevador descer" << endl;
11    cout << "3) para entrar no Elevador" << endl;
12    cout << "4) para sair do Elevador" << endl;
13    cout << "5) para mostrar o status" << endl;
14    cout << "6) para sair do programa" << endl;
15 }
16
```

## elevador.h

```
1  #ifndef ELEVADOR_H
2  #define ELEVADOR_H
3
4  #include<iostream>
5  #include<string>
6
7  using namespace std;
8
9  #define TERREO 0
10 #define VAZIO 0
11
12 class Elevador{
13 private:
14     int andarAtual;
15     int totalAndares;
16     int capacidade;
17     int lotacao;
18
19 public:
20     Elevador();
21     Elevador(int andarAtual, int totalAndares, int capacidade, int lotacao);
22
23     void inicializa(int capacidade, int totalAndares);
24     bool entra();
25     bool sai();
26     bool sobe();
27     bool desce();
28     void mostrar();
29     string toString();
30
31     void setAndarAtual(int andarAtual);
32     void setTotalAndares(int totalAndares);
33     void setCapacidade(int capacidade);
34     void setLotacao(int lotacao);
35     int getAndarAtual();
36     int getTotalAndares();
37     int getCapacidade();
38     int getLotacao();
39 };
40
41 #endif // !ELEVADOR_H
42
43
```

## elevador.cpp

```
1 #include "elevador.h"
2
3 Elevador::Elevador(){
4     this->andarAtual = 0;
5     this->lotacao = 0;
6     this->capacidade = 0;
7     this->totalAndares = 0;
8 }
9
10 Elevador::Elevador(int andarAtual, int lotacao, int capacidade, int totalAndares){
11     this->andarAtual = andarAtual;
12     this->lotacao = lotacao;
13     this->capacidade = capacidade;
14     this->totalAndares = totalAndares;
15 }
16
17 void Elevador::inicializa(int capacidade, int totalAndares){
18     this->andarAtual = TERREO;
19     this->lotacao = VAZIO;
20     this->capacidade = capacidade;
21     this->totalAndares = totalAndares;
22 }
23
24 bool Elevador::entra(){
25     bool resp = false;
26     if(lotacao + 1 <= capacidade){
27         lotacao ++;
28         resp = true;
29     }
30     return resp;
31 }
32
33 bool Elevador::sai(){
34     bool resp = false;
35     if (lotacao >= 1){
36         lotacao--;
37         resp = true;
38     }
39     return resp;
40 }
41
42 bool Elevador::sobe(){
43     bool resp = false;
44     if (andarAtual + 1 <= totalAndares){
45         andarAtual++;
46         resp = true;
47     }
48     return resp;
49 }
50
51 bool Elevador::desce(){
52     bool resp = false;
53     if (andarAtual-1 >= TERREO){
54         andarAtual--;
55         resp = true;
56     }
57     return resp;
58 }
59
60 void Elevador::mostrar(){
61     cout << "Andar Atual: " << andarAtual << endl;
62     cout << "Lotacao: " << lotacao << endl;
63     cout << "Capacidade: " << capacidade << endl;
64     cout << "Total Andares: " << totalAndares << endl;
65 }
```

```
66
67 string Elevador::toString(){
68     string resp = "";
69     resp = "Andar Atual: " + andarAtual;
70     resp += " Lotacao: " + lotacao;
71     resp += " Capacidade: " + capacidade;
72     resp += " Total Andares: " + totalAndares;
73     return resp;
74 }
75
76 void Elevador::setAndarAtual(int andarAtual){
77     this->andarAtual = andarAtual;
78 }
79
80 void Elevador::setTotalAndares(int totalAndares){
81     this->totalAndares = totalAndares;
82 }
83
84 void Elevador::setCapacidade(int capacidade){
85     this->capacidade = capacidade;
86 }
87
88 void Elevador::setLotacao(int lotacao){
89     this->lotacao = lotacao;
90 }
91
92 int Elevador::getAndarAtual(){
93     return andarAtual;
94 }
95
96 int Elevador::getTotalAndares(){
97     return totalAndares;
98 }
99
100 int Elevador::getCapacidade(){
101     return capacidade;
102 }
103
104 int Elevador::getLotacao(){
105     return lotacao;
106 }
107
```