

```

1 // Programação Orientada a Objectos 2020/2021
2 // Ana Rita Videira - 5012012218
3
4 //Exame 1920 - Época Normal - Exercício 7
5
6
7 #include <cstdlib>
8 #include <iostream>
9 #include <sstream>
10 #include <fstream>
11 #include <string>
12 #include <ctype.h>
13 #include <vector>
14
15 using namespace std;
16
17 class Funcionario {
18
19 public:
20     Funcionario(const string &n) : nome(n) {};
21     virtual string getNome() const { return nome; };
22     virtual string cumprimentar() const = 0;
23
24 protected:
25     string nome;
26 };
27
28
29 class Doutor : public Funcionario{
30
31 public:
32
33     Doutor(const string &no) : Funcionario(no) { };
34
35     string cumprimentar() const override {
36
37         ostringstream os;
38         os << "Bom dia Doutor " << getNome() << endl;
39         return os.str();
40     }
41
42 };
43
44 class Engenheiro : public Funcionario {
45
46 public:
47     Engenheiro(const string &no) : Funcionario(no) { };
48
49     string cumprimentar() const override {
50
51         ostringstream os;
52         os << "Bom dia Engenheiro " << getNome() << endl;
53         return os.str();
54     }
55
56 };
57
58 class Empresa {
59
60 public:
61
62     Empresa() {
63         funcionarios.push_back(new Doutor("D1"));
64         funcionarios.push_back(new Doutor("D2"));
65         funcionarios.push_back(new Engenheiro("E1"));
66         funcionarios.push_back(new Engenheiro("E2"));
67     }
68
69
70     void cumprimentar() const{
71         for(auto a : funcionarios)
72             cout << a->cumprimentar();
73     }

```

```
74
75 private:
76     vector<Funcionario*>funcionarios;
77
78
79 };
80
81
82
83 int main(int argc, char** argv) {
84
85     Empresa empresa;
86     empresa.cumprimentar();
87
88     return 0;
89 }
90
91
```