

```

1 // Programação Orientada a Objectos 2020/2021
2 // Ana Rita Videira - 5012012218
3
4 //Exame 1920 - Época Normal - Exercício 6
5
6 #include <stdio.h>
7 #include <vector>
8 #include <string>
9 #include <sstream>
10 #include <iostream>
11
12 using namespace std;
13
14 class Bilhete {
15 public:
16
17     //Criar objectos indicando todos os dados inclusive um conjunto inicial de malas
18
19     Bilhete(string p, int pa, string &co, vector<int> ma)
20     : passageiro(p), passaporte(pa), companhia(co), id_malas(ma) { }
21
22     // Bilhete(const Bilhete& orig);
23     // virtual ~Bilhete();
24
25
26     // atribuir objectos -> operador atribuicao ( = )
27
28     Bilhete &operator=(const Bilhete &b) {
29         passageiro = b.passageiro;
30         passaporte = b.passaporte;
31
32         for (size_t i = 0; i < b.id_malas.size(); i++)
33             id_malas.push_back(b.id_malas[i]);
34
35         return *this;
36     }
37
38
39
40     //Acrescentar bagagens ao bilhete
41     //(representadas pelos seus id e sem repetir)
42     // bilhete << 123 << 789 << 123; -> ignora 123
43
44     Bilhete &operator<<(int p) {
45
46         for (auto i : id_malas) {
47
48             if (p == id_malas[i])
49                 return *this;
50         }
51
52         id_malas.push_back(p);
53
54         return *this;
55     }
56
57
58
59
60     // Remover todas as bagagens com id superior a um indicado
61     // (bilhete2 -=40 ) -=35;
62
63     Bilhete &operator--(int id) {
64
65         auto it = id_malas.begin();
66
67         while (it != id_malas.end()) {
68
69             if (id >= *it)
70                 it = id_malas.erase(it);
71             else
72                 it++;
73

```

```

74         }
75
76         return *this;
77     }
78
79
80     string getAsString() const {
81         ostream os;
82
83         os << "Passageiro: " << passageiro << " Passaporte: " << passaporte << "
            Companhia: "
84             << companhia << " Malas: ";
85         for (auto i : id_malas)
86             os << " " << i;
87
88         return os.str();
89     }
90
91
92 private:
93
94     string passageiro;
95     int passaporte;
96     string & companhia;
97     vector <int> id_malas;
98
99 };
100
101
102
103 // Mostrar Conteudo no ecrã da forma
104 // cout << "\nPassageiro 1: " << bilhete1 << "Passageiro 2 << bilhete 2;
105
106 ostream &operator<<(ostream &os, const Bilhete &b) {
107     return os << b.getAsString();
108 }
109
110
111 int main(int argc, char** argv) {
112
113     string ola = "ola";
114
115     Bilhete bi("Bruno", 123, ola,{1, 30, 3, 40, 5});
116     Bilhete a("Rafael", 123, ola,{3, 4});
117
118     cout << bi << endl;
119
120     bi << 123 << 789 << 123;
121
122     (bi -= 20) -= 30;
123
124     cout << bi;
125
126     return 0;
127 }
128
129

```