

```

1 //-----
2 // Nom           : Exercice 5.25
3 // Fichier        : main.cpp
4 // Version        : 01 - 29-11-2022
5 // Auteur(s)      : HÄFFNER EDWIN
6 // But            : Vérifier si deux tableaux sont égaux (au sens ensembliste du terme)
7 // Modifications :
8 // Remarque(s)   :
9 //-----
10
11 #include <iostream>
12 #include <vector>
13 #include <limits>
14 #include <algorithm>
15
16 using namespace std;
17 using Vecteur = vector<int>;
18
19 ostream &operator<<(ostream &os, const Vecteur &v);
20
21 bool estEgal(Vecteur &v1, Vecteur &v2);
22
23 int main() {
24
25     Vecteur v1 = {1, 2, 3, 4, 5};
26     Vecteur v2 = {1, 3, 4, 5, 2, 2};
27
28     cout << "Vecteur 1" << endl
29           << v1 << endl
30           << "Vecteur 2" << endl
31           << v2 << endl << endl;
32
33     cout << "Les deux vecteur sont ";
34     if (estEgal(v1,v2)){
35         cout << "egaux" << endl;
36     } else {
37         cout << "inegaux" << endl;
38     }
39
40     cin.ignore(numeric_limits<streamsize>::max(), '\n');
41
42     return 0;
43 }
44
45 ostream &operator<<(ostream &os, const Vecteur &v) {
46     cout << "[";
47     for (auto it = v.begin(); it != v.end(); ++it) {
48         os << *it;
49         if (it < v.end() - 1)
50             os << ", ";
51     }
52     cout << "]";
53     return os;
54 }
55
56 bool estEgal(Vecteur &v1, Vecteur &v2) {
57
58     sort(v1.begin(), v1.end());
59     sort(v2.begin(), v2.end());
60     auto v1u = unique(v1.begin(), v1.end());
61     auto v2u = unique(v2.begin(), v2.end());
62     v1.erase(v1u, v1.end());
63     v2.erase(v2u, v2.end());
64
65     return equal(v1.begin(), v1.end(), v2.begin());
66
67 }

```