

```
1: // $Id: printvector.cpp,v 1.7 2022-01-12 17:25:06-08 - - $
2:
3: #include <iostream>
4: #include <vector>
5: using namespace std;
6:
7: // Example of printing a vector of uint8_t.
8:
9: ostream& operator<< (ostream& out, const vector<uint8_t>& vec) {
10:     for (uint8_t byte: vec) out << static_cast<int> (byte);
11:     return out;
12: }
13:
14: void print_rev1 (const vector<uint8_t>& vec) {
15:     for (auto itor = vec.crbegin(); itor != vec.crend(); ++itor) {
16:         cout << static_cast<int> (*itor);
17:     }
18: }
19:
20: void print_rev2 (const vector<uint8_t>& vec) {
21:     // Note: ssize_t, not size_t, because must stop at -1, not 0.
22:     for (ssize_t index = vec.size() - 1; index >= 0; --index) {
23:         cout << static_cast<int> (vec[index]);
24:     }
25: }
26:
27: int main() {
28:     vector<uint8_t> data {6, 3, 8, 9, 0, 2, 1};
29:     cout << "operator<<: " << data << endl;
30:     cout << "print_rev1: "; print_rev1 (data); cout << endl;
31:     cout << "print_rev2: "; print_rev2 (data); cout << endl;
32: }
33:
34: //TEST// ./printvector >printvector.out
35: //TEST// mkpspdf printvector.ps printvector.cpp printvector.out
36:
```

01/12/22
17:25:08

\$cse111-wm/Assignments/asg1-dc-bigint/misc
printvector.out

1/1

```
1: operator<<: 6389021
2: print_rev1: 1209836
3: print_rev2: 1209836
```