

```
#include <iostream>
#include <iterator>
#include <vector>
#include <memory>

using namespace std;

int main() {

    vector<int> numbers {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

    reverse_iterator<vector<int>::iterator> begin (numbers.end());
    reverse_iterator<vector<int>::iterator> end(numbers.begin());

    while( begin != end) {
        cout << *begin++ << " ";
    }

    cout << endl;

    vector<int>::iterator it = numbers.begin();
    std::advance(it, 3);

    insert_iterator<vector<int>> inserter(numbers, it);

    for (int i = 1; i <= 5; ++i) {
        *inserter = (i*10);
    }

    cout << "numbers: ";
    for (auto i : numbers) {
        cout << i << " ";
    }

    cout << endl;

    vector<unique_ptr<int>> pointers;
    vector<unique_ptr<int>> others;

    using iter_t = vector<unique_ptr<int>>::iterator;

    for (int i = 0; i < 10; ++i) {
        int* tmp = new int;
        *tmp = i;
        pointers.push_back(unique_ptr<int>(tmp));
    }

    cout << "pointers size: " << pointers.size() << " others size: " << others.size() <<
    "\n";

    back_insert_iterator<vector<unique_ptr<int>>> bininsert(others);

    copy(move_iterator<iter_t>(pointers.begin()), move_iterator<iter_t>(pointers.end()),
    bininsert);

    pointers.clear(); //They are now undefined after the move

    cout << "pointers size: " << pointers.size() << " others size: " << others.size() <<
    "\n";

    return 0;
}
```