#include<iostream>

using namespace std;

class Iterator {

int\* ptr;

public:

Iterator():ptr(NULL){}

Iterator(int\*ptr):ptr(ptr){}

Iterator& operator++() {

ptr++;

return \*this;

}

Iterator& operator++(int) {

ptr++;

return \*this;

}

bool operator!=(const Iterator& other) {

return this->ptr != other.ptr;

}

bool operator==(const Iterator& other) {

return this->ptr == other.ptr;

}

int& operator\*() {

return \*ptr;

}

};

class Array {

int\* data;

int size = 0;

public:

Array(int size) : size(size) {

data = new int[size] {};

for (size\_t i = 0; i < size; i++)

{

data[i] = rand() % 30;

}

}

Iterator begin() {

return Iterator(data);

}

Iterator end() {

return Iterator(data + size);

}

};

#include<vector>

void main() {

//vector<int>v;

//for (size\_t i = 0; i < 5; i++)

//{

// v.push\_back(i + 1);

//}

//for (auto i : v) {

// cout << i << endl;

//}

Array a(10);

for (auto i = a.begin(); i!=a.end(); i++)

{

cout << \*i << endl;

}

for (auto i : a) {

cout << i << endl;

}

}