NAME LAIBA

BCS231040

QUESTION NUMBER 2

//inventory

#include <iostream>

#include <vector>

#include <algorithm>

#include <string>

using namespace std;

struct Product {

int id;

string name;

double price;

};

vector<Product> inventory;

void addProduct(int id, const string& name, double price) {

for (const auto& product : inventory) {

if (product.id == id) {

cout << "Product with ID " << id << " already exists in the inventory"<<endl;

return;

}

}

inventory.push\_back({id, std::string(name), price});

cout << "Product added to the inventory"<<endl;

}

void removeProduct(int id) {

auto it = remove\_if(inventory.begin(), inventory.end(),

[id](const Product& product) { return product.id == id; });

if (it != inventory.end()) {

inventory.erase(it, inventory.end());

cout << "Product with ID " << id << " removed from the inventory."<<endl;

} else {

cout << "Product with ID " << id << " not found in the inventory"<<endl;

}

}

int main() {

addProduct(1, "Laptop", 999.99);

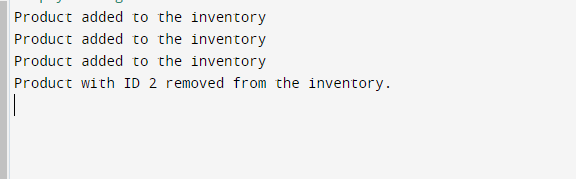
addProduct(2, "Smartphone", 499.99);

addProduct(3, "Tablet", 299.99);

removeProduct(2);

return 0;

}



//part 2

#include <iostream>

#include <vector>

#include <algorithm>

#include <chrono>

using namespace std;

int main() {

vector<int> numbers(100000);

for (int i = 0; i < 100000; ++i) {

numbers[i] = 100000 - i;

}

// Bubble Sort

auto startTimeBubbleSort = chrono::high\_resolution\_clock::now();

sort(numbers.begin(), numbers.end());

auto endTimeBubbleSort = chrono::high\_resolution\_clock::now();

auto durationBubbleSort = chrono::duration\_cast<chrono::microseconds>(endTimeBubbleSort - startTimeBubbleSort);

cout << "Bubble Sort Execution Time: " << durationBubbleSort.count() << " microseconds"<<endl;

// Print 10 integers

cout << "First 10 integers: ";

for (int i = 0; i < 10; ++i) {

cout << numbers[i] << " ";

}

cout << endl;

cout << "Last 10 integers: ";

for (int i = 99990; i < 100000; ++i) {

cout << numbers[i] << " ";

}

cout << endl;

return 0;

}

