#include <iostream>

#include <vector>

#include <limits>

#include <omp.h>

int main() {

std::vector<int> arr = {5, 2, 8, 12, 1, 6, 3, 9};

int minVal = std::numeric\_limits<int>::max();

int maxVal = std::numeric\_limits<int>::min();

int sum = 0;

int count = 0;

#pragma omp parallel for reduction(min:minVal) reduction(max:maxVal) reduction(+:sum) reduction(+:count)

for (int i = 0; i < arr.size(); i++) {

minVal = std::min(minVal, arr[i]);

maxVal = std::max(maxVal, arr[i]);

sum += arr[i];

count++; }

double average = static\_cast<double>(sum) / count;

std::cout << "Min: " << minVal << std::endl;

std::cout << "Max: " << maxVal << std::endl;

std::cout << "Sum: " << sum << std::endl;

std::cout << "Avg: " << average << std::endl;

return 0;}