HPC-3(Min, Max)

#include <iostream>

#include <omp.h>

#include <vector>

Using namespace std;

Int main() {

Int n;

Cout << “Enter number of elements: “;

Cin >> n;

Vector<int> arr(n);

Cout << “Enter “ << n << “ elements:\n”;

For (int I = 0; I < n; ++i)

Cin >> arr[i];

Int minVal = arr[0];

Int maxVal = arr[0];

Int sum = 0;

#pragma omp parallel for reduction(min:minVal)

For (int I = 0; I < n; ++i) {

If (arr[i] < minVal)

minVal = arr[i];

}

#pragma omp parallel for reduction(max:maxVal)

For (int I = 0; I < n; ++i) {

If (arr[i] > maxVal)

maxVal = arr[i];

}

#pragma omp parallel for reduction(+:sum)

For (int I = 0; I < n; ++i) {

Sum += arr[i];

}

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

Cout << “\nMinimum: “ << minVal << endl;

Cout << “Maximum: “ << maxVal << endl;

Cout << “Sum : “ << sum << endl;

Cout << “Average: “ << average << endl;

Return 0;

}