

CSE4001 – Lab Assessment 2

17BCI0113 – Namit Nathwani

Prof. Jaisakthi S M – L55+L56

Question: Write a simple OpenMP program to employ a 'reduction' clause to express the reduction of a for loop.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include <omp.h>

using namespace std;

void doSum(double *A, int n, int threads) {
    omp_set_dynamic(0);
    double sum = 0;

    #pragma omp parallel for num_threads(threads) reduction(+:
sum)
        for(int i = 0; i < n; i++)
            sum += A[i];

    cout << "Sum = " << sum << endl;
}

int main() {
    int i, n, threads;
    double *A, dtime, sum;
```

```

n = 100000000;

A =(double*) malloc(sizeof(double) * n);

for(i = 0; i < n; i++)
    A[i] = 1.0 * rand()/RAND_MAX;

cout << "n = " << n << endl;

threads = 8;
dtime = omp_get_wtime();
doSum(A, n, threads);
dtime = omp_get_wtime() - dtime;

cout << "Threads = " << threads << endl;
cout << "Time Taken = " << dtime << "s\n" << endl;

threads = 4;
dtime = omp_get_wtime();
doSum(A, n, threads);
dtime = omp_get_wtime() - dtime;

cout << "Threads = " << threads << endl;
cout << "Time Taken = " << dtime << "s\n" << endl;

threads = 2;
dtime = omp_get_wtime();
doSum(A, n, threads);
dtime = omp_get_wtime() - dtime;

cout << "Threads = " << threads << endl;
cout << "Time Taken = " << dtime << "s\n" << endl;

```

```

    threads = 1;

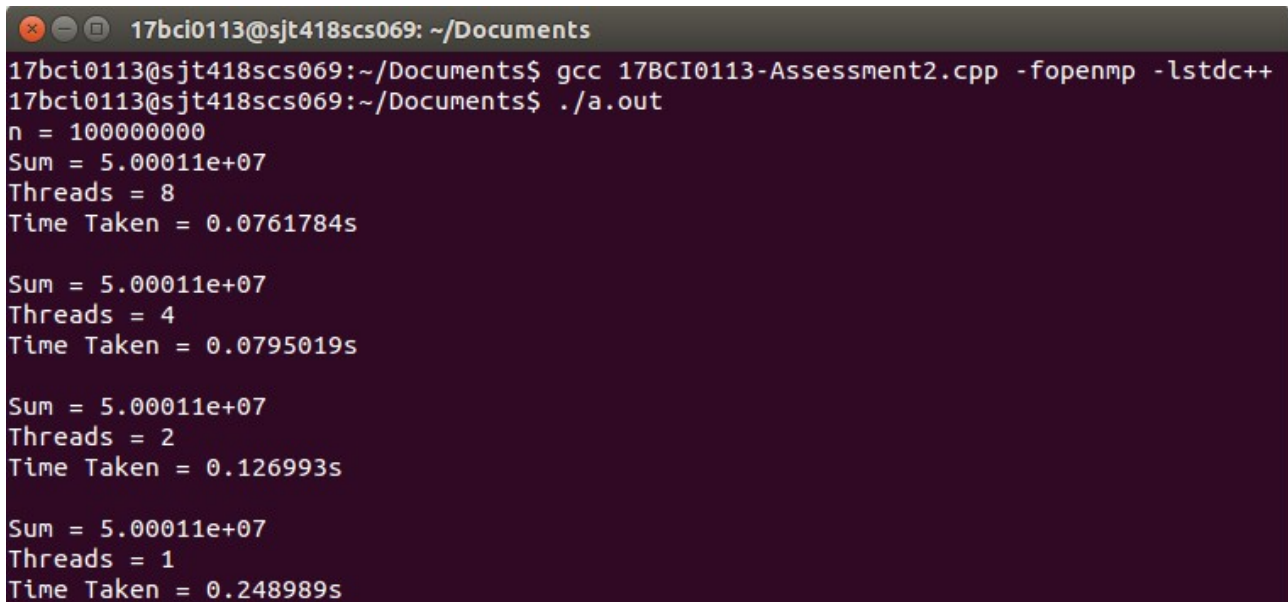
    dtime = omp_get_wtime();
    doSum(A, n, threads);
    dtime = omp_get_wtime() - dtime;

    cout << "Threads = " << threads << endl;
    cout << "Time Taken = " << dtime << "s\n" << endl;

}

```

Output:



```

17bci0113@sjt418scs069: ~/Documents
17bci0113@sjt418scs069:~/Documents$ gcc 17BCI0113-Assessment2.cpp -fopenmp -lstdc++
17bci0113@sjt418scs069:~/Documents$ ./a.out
n = 100000000
Sum = 5.00011e+07
Threads = 8
Time Taken = 0.0761784s

Sum = 5.00011e+07
Threads = 4
Time Taken = 0.0795019s

Sum = 5.00011e+07
Threads = 2
Time Taken = 0.126993s

Sum = 5.00011e+07
Threads = 1
Time Taken = 0.248989s

```