COL 380
Assignment 1

Submitted by:-
Gaurav Chauhan
2018CS50406

The two approaches are used as discussed in the class:-

1) The first approach is the one in which we divide the sum into groups and each thread works into computing the value of the given group, after which the master thread computes the sum of the values obtained by the individual threads.

2) The second approach is the one in which we form a tree like structure and parallelly compute the sum of the given two continuous value in a level of the tree.

Compilation Instructions:

Step 1:

```
gcc -O0 -fopenmp 2018CS50406.c
```

Step 2:

```
./a.out (Approach 1 or 2) (Number of threads) (N)
```

Time taken for:

Approach 1

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.000011	0.000924	0.037145
4	0.000008	0.000449	0.040998
8	0.000005	0.000331	0.036035

Approach 2

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.000027	0.000459	0.044193
4	0.000028	0.000251	0.043798
8	0.000030	0.000145	0.031032

NOTE:-

Here these values correspond to the median of 50 values, this is done to prevent the large deviations.

Serial	N = 1000	N = 100,000	N = 10,000,000
	0.000007	0.000532	0.046231

Speed Ups:

Approach 1

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.68	0.59	1.06
4	1.00	1.15	1.08
8	1.34	1.62	1.17

Approach 2

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.28	1.13	1.03
4	0.27	2.10	1.04
8	0.27	3.34	1.46

Efficiency:

Approach 1

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.34	0.29	0.59
4	0.23	0.30	0.29
8	0.17	0.21	0.15

Approach 2

Threads	N = 1000	N = 100,000	N = 10,000,000
2	0.13	0.57	0.51
4	0.08	0.52	0.27
8	0.03	0.43	0.18

Amdahl’s Law→

f = (1 - 1/S) * (1 - 1/p)

Approach 1

Threads	N = 1000	N = 100,000	N = 10,000,000
2	-1.00	-1.54	0.34
4	0.01	0.57	0.35
8	2.00	3.10	1.28

Approach 2

Threads	N = 1000	N = 100,000	N = 10,000,000
2	-5.10	0.27	0.08
4	-10.60	2.12	0.11
8	-21.00	5.43	2.58

We see that the value of f is not constant and hence the amdahl’s law is not followed. This is because the law does not consider the overhead computations caused due to processes needed to provide for parallelism.