Lab Submission – 05

Arnab Mondal

20BCE1294

Program: B.Tech

Semester: Fall 2022-23

Course: CSE4001 – Parallel and Distributed Computing

Faculty: Dr. Sudha A

Date: 07-09-2022

Exercise: 05

1. Do the Combined and Orphaned parallel loop reduction for marks total of one subject for 70 students.

   Code:

```c
#include <omp.h>
#include <stdio.h>

void cal(int arr[])
{
    int sum = 0;
    double st, en;
    st = omp_get_wtime();
    #pragma omp parallel for
    for(int i=0;i<70;i++)
        {
            sum += arr[i];
        }
    en = omp_get_wtime();
    printf("Time : %f\n", (en - st));
    printf("Sum : %d\n", sum);
}
```

```c
int main()
{
    int arr[70], opt;
    double st, en;
    printf("\n1.Normal loop sharing\n2.Orphaned process\nEnter option : ");
    scanf("%d", &opt);
    for(int i=0;i<70;i++)
    {
        arr[i] = i;
    }
    int sum = 0;
    if(opt == 1)
    {
        st = omp_get_wtime();
        #pragma omp parallel for
        for(int i=0;i<70;i++)
            {
                sum += arr[i];
            }
        en = omp_get_wtime();
        printf("Time : %f\n", (en - st));
        printf("Sum : %d\n", sum);
    }
    else if(opt == 2)
    {
        cal(arr);
    }
    return 0;
}
```

Output:

```
codebind@arnabmondal20bce1294: ~/ArnabMondal20BCE1294/CSE4001_PDC
File  Edit  View  Search  Terminal  Help
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ gedit lab5_1.c

codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ gcc -fopenmp l
ab5_1.c
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ ./a.out

1.Normal loop sharing
2.Orphaned process
Enter option : 1
Time : 0.010952
Sum : 2415
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ ./a.out

1.Normal loop sharing
2.Orphaned process
Enter option : 2
Time : 0.011097
Sum : 2261
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$
```

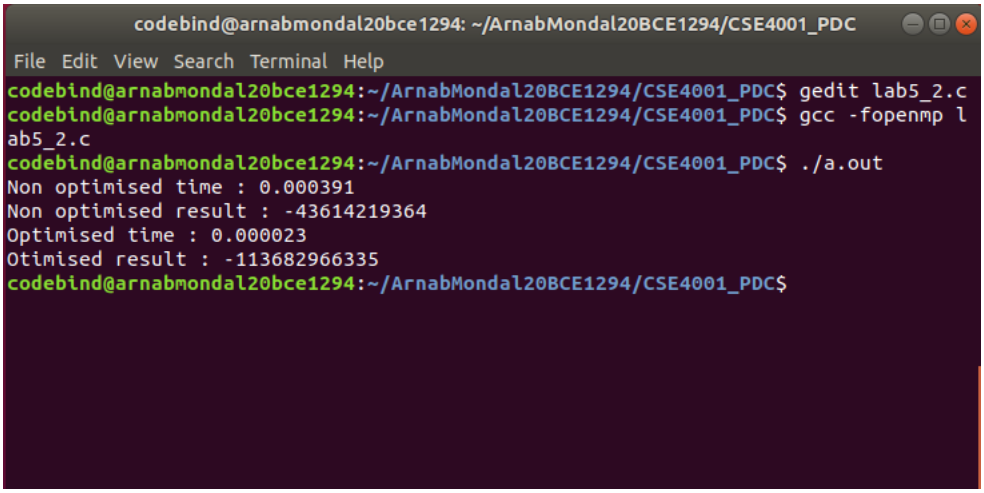
2. Show one optimization Technique

Code:

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

int main()
{
    int arr[100], opt;
    double st, en;
    for(int i=0;i<70;i++)
    {
        arr[i] = 50;
    }
    long res = 0;
    int x=100, y=2000, z=10750;
    int temp1 = (x*y*z) / 3;
    int temp2 = (x+y+z) / 7;
    temp1 += temp2;
    st = omp_get_wtime();
    #pragma omp parallel for
    for(int i=0;i<100;i++)
        {
            res += (((x*y*z) / 3) + ((x+y+z) / 7)) + i;
        }
    en = omp_get_wtime();
    printf("Non optimised time : %f\n", (en - st));
    printf("Non optimised result : %ld\n", res);
    st = omp_get_wtime();
    #pragma omp parallel for
    for(int i=0;i<100;i++)
        {
            res += temp1 + i;
        }
    en = omp_get_wtime();
    printf("Optimised time : %f\n", (en - st));
    printf("Otimised result : %ld\n", res);
    return 0;
}
```

Output:


codebind@arnabmondal20bce1294: ~/ArnabMondal20BCE1294/CSE4001_PDC
File Edit View Search Terminal Help
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ gedit lab5_2.c
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ gcc -fopenmp l
ab5_2.c
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$ ./a.out
Non optimised time : 0.000391
Non optimised result : -43614219364
Optimised time : 0.000023
Otimised result : -113682966335
codebind@arnabmondal20bce1294:~/ArnabMondal20BCE1294/CSE4001_PDC$