

## OPENMP – MONTE CARLO

## EXTRA QUESTION

**Aim:** To calculate the value of “pi = 3.14” using the Monte Carlo method.

CODE AND OUTPUTS

## 1. Monte Carlo – CODE WITH OpenMP

```

1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<omp.h>
4 int main()
5 {
6     int i,count;
7     unsigned short a[3];
8     int samp;
9     double x,y;
10    double pi;
11    printf("\n ----- ");
12    printf("\n\tMONTE CARLO: ESTIMATION OF PI");
13    printf("\n ----- ");
14    printf("\n\tENTER THE SAMPLE SIZE: ");
15    scanf("%d",&samp);
16    printf("\n");
17
18    #pragma omp parallel
19    {
20        a[0]=1;
21        a[1]=1;
22        a[2]=omp_get_thread_num();
23        count=0;
24        printf("\tThread executing: %d\n",a[2]);
25
26        #pragma omp for firstprivate(a) private(x,y) reduction(+:count)
27        for(i=0; i<samp; i++)
28        {
29            x=erand48(a);
30            y=erand48(a);
31            if(x*x+y*y<=1.0)count++;
32        }
33    }
34    pi=4.0*(double)count/(double)samp;
35    printf("\n\tCOUNT : %d, \tSAMPLE SIZE : %d,\n\tESTIMATED VALUE OF PI : %.5f\n\n", count, samp, pi);
36 }

```

## OUTPUT

For sample size = 2000

```

dhruv@dhruv-Inspiron-5559:~/PDC-lab/Monte Carlo$ ./omp_monteCarlo

ENTER THE SAMPLE SIZE: 2000

Thread executing: 0
Thread executing: 0
Thread executing: 2
Thread executing: 3

COUNT : 1484,    SAMPLE SIZE : 2000,
ESTIMATED VALUE OF PI : 2.96800

```

For sample size = 25000

```
dhruv@dhruv-Inspiron-5559:~/PDC-lab/Monte Carlo$ ./omp_monteCarlo

ENTER THE SAMPLE SIZE: 25000

Thread executing: 2
Thread executing: 0
Thread executing: 1
Thread executing: 3

COUNT : 19552, SAMPLE SIZE : 25000,
ESTIMATED VALUE OF PI : 3.12832
```

For sample size = 250000

```
dhruv@dhruv-Inspiron-5559:~/PDC-lab/Monte Carlo$ ./omp_monteCarlo

ENTER THE SAMPLE SIZE: 250000

Thread executing: 0
Thread executing: 2
Thread executing: 3
Thread executing: 1

COUNT : 196232, SAMPLE SIZE : 250000,
ESTIMATED VALUE OF PI : 3.13971
```

### INFERENCE

As we can see, the accuracy of the estimated pi value increases on increasing the sample size, when implementing the Monte Carlo approach.