Arpit Kumar
1RV17CS024
Batch - A1

Program- Monte Carlo Pi Calculation

```
#include<stdio.h>
#include<stdlib.h>
#include<omp.h>
#define SEED 50123

int main()
{
	long n = 0, i, count = 0;
	double x,y,z;


//	printf("Enter the number of iterations used to estimate pi: ");
//	scanf("%ld", &n);
	srand(SEED);
	printf("Size\t\tT1\t\tT2\t\tT4\t\tT8");
	for(n=10; n<=1000000; n*=10){
		printf("\n%ld\t",n);
		for(int t=1; t<=8; t*=2){
			count = 0;
			double start = omp_get_wtime();
			omp_set_num_threads(t);

			#pragma omp parallel for private(x, y, z) shared(count)
			for ( i=0; i<n; i++)
			{
				x = (double)rand()/RAND_MAX;
				y = (double)rand()/RAND_MAX;
				z = x*x+y*y;
				if (z<=1) count++;
			}

			double pi=(double)count/n * 4;
			double stop = omp_get_wtime();
			printf("%lf %lfs\t",pi,stop-start);
		}
	}
	return 0;
}
```

Output-

```
rohit@Rohit: /mnt/c/users/rohit/Desktop
rohit@Rohit:/mnt/c/users/rohit/Desktop$ cc -fopenmp pi.c
rohit@Rohit:/mnt/c/users/rohit/Desktop$ ./a.out
Size            T1                      T2                      T4                      T8
10      2.800000 0.000026s      3.200000 0.000279s      3.600000 0.000289s      2.800000 0.000381s
100     3.120000 0.000009s      3.120000 0.000125s      2.880000 0.000323s      3.240000 0.000426s
1000    3.160000 0.000043s      3.184000 0.000084s      3.100000 0.000437s      3.096000 0.000586s
10000   3.127200 0.000388s      3.169200 0.001391s      3.145600 0.002167s      3.106800 0.002289s
100000  3.137840 0.003848s      3.135600 0.012617s      3.124920 0.020626s      3.123080 0.020887s
rohit@Rohit:/mnt/c/users/rohit/Desktop$
```

Monte Carlo Graph: