

DTS202TC Foundation of Parallel Computing

Lab 4: Shared-Memory Programming with OpenMP

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <omp.h>
4
5  void Hello(void); /* Thread function */
6
7  int main(int argc, char* argv[]) {
8      /* Get number of threads from command line */
9      int thread_count = strtol(argv[1], NULL, 10);
10
11     # pragma omp parallel num_threads(thread_count)
12     Hello();
13
14     return 0;
15 } /* main */
16
17 void Hello(void) {
18     int my_rank = omp_get_thread_num();
19     int thread_count = omp_get_num_threads();
20
21     printf("Hello from thread %d of %d\n", my_rank, thread_count);
22
23 } /* Hello */
```

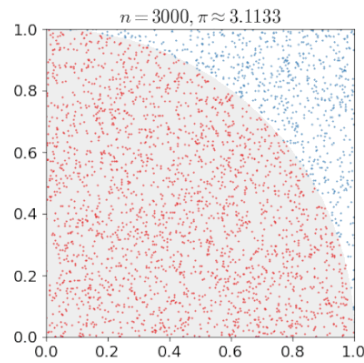
Figure 1: OpenMP HelloWorld

OpenMP Hello World

Type the above HelloWorld source code manually in a editor,

Compile and run the program 3 times and observe the results. Search for the api that prints out the maximum number of threads available, put it in the code above so that it prints out the value.

Monte-Carlo Estimation with OpenMP



Suppose we toss darts randomly at a square dartboard, whose bullseye is at the origin, and whose sides are 2 feet in length. Suppose also that there's a circle inscribed in the square dartboard. The radius of the circle is 1 foot, and it's area is π square feet. If the points that are hit by the darts are uniformly distributed (and we always hit the square), then the number of darts that hit inside the circle should approximately satisfy the equation

$$\frac{\text{number in circle}}{\text{total number of tosses}} = \frac{\pi}{4} \quad (1)$$

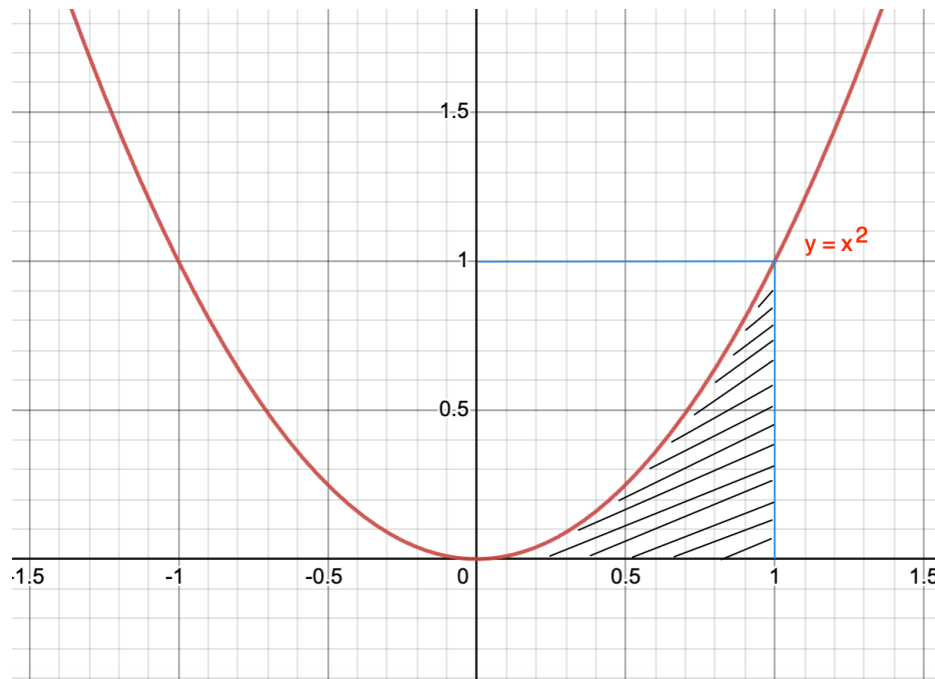


Figure 2: Estimate the shaded area

This is called a “Monte Carlo” method, since it uses randomness (the dart tosses). Write an OpenMP program that uses a Monte Carlo method to estimate the shaded area in Figure 2. Read in the total number of tosses before forking any threads. Print the result after joining all the threads. You may want to use long long ints for the number of hits and the number of tosses, since both may have to be very large to get a reasonable estimation.

Comparison

Compare and analyse the performances of the Monte-Carlo Estimation programs from lab 3 and 4 on both efficiency and speedup.

Task 4, Reflection

What challenge did you face and how did you overcome it?