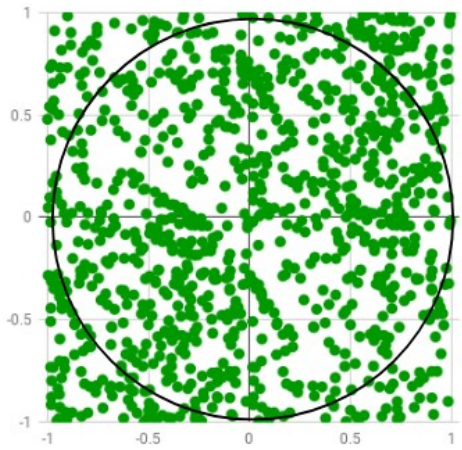
CENG444 **Spring 2023**

PROGRAMMING ASSIGNMENT 1

Due date: 19.04.2023 23:00

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 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

$$\frac{\text{number in circle}}{\text{total number of tosses}} = \frac{\pi}{4},$$

since the ratio of the area of the circle to the area of the square is= 4. We can use this formula to estimate the value of with a random number generator:

```
number_in_circle = 0;
for (toss = 0; toss < number_of_tosses; toss++) {
   x = random double between −1 and 1;
   y = random double between −1 and 1;
   distance_squared = x*x + y*y;
   if (distance_squared <= 1) number_in_circle++;
}
pi_estimate = 4*number_in_circle/((double) number_of_tosses);
```

This is called a "Monte Carlo" method, since it uses randomness (the dart tosses).

You are required to develop a parallel version of this program on OpenMP that uses Monte Carlo method to estimate pi.

Notes:

- You may want to use ***long long int*** for the number of hits in the circle and the number of tosses, since both may have to be very large to get a reasonable estimate of pi. You can try with different number of tosses and report your results for only one toss value.

- You don't need to try to optimize the performance, however, in your submission, you can include optimized versions as well if you have any.

- You are required to run your programs with various number of threads (1-16) and compare different versions.

- You are required to write a report (2 pages at most), which includes
  - Your computer's specification (i.e., the number of cores, the number of threads; typically that can be obtained by *lscpu* command in Linux platform),
  - Instructions for compiling and executing your programs,
  - Graphs and your observations about the performance of your implementation.

Submission: You are required to submit your **commented** source code and report to cloud-lms. Please create a compressed file including all source files and report; and name it as yourstudentnumber.zip (e.g. If your student number is 202312345678, the file name must be 202312345678.zip). You need to work individually, no group work is allowed. No late submission will be accepted.