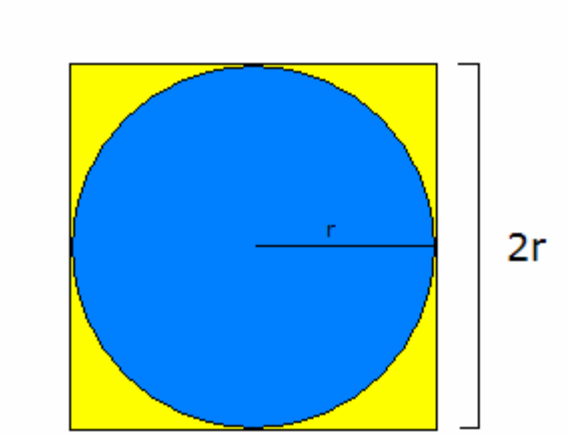
CDA 5106 Spring 2009
Homework 1 (Chapter 1)

Due Tue, Jan 20th in class

Q1. [Amdahl's Law, parallelization] 30 pts

Consider one of the methods for approximating pi. The first step is to inscribe a circle inside a square:



We can approximate pi by generating random points and *counting* those falling within a circle. (i.e., for 2 random numbers x and y, a number is in a circle if $x^2 + y^2 < r^2$).

a) Write a parallel algorithm that performs the above computation given N worker processes.

b) If the number of clock cycles per specific instruction is given below (on a hypothetical computer), derive a formula for computing the algorithm's *speedup* given N worker processes:

- Multiply, Divide, Add, Subtract: 5 CPI (each)
- Random number generation: 3 CPI
- Comparison: 2 CPI
- Conditional Branch (incl. comparison): 2 CPI
- Inter-process communications (submit task, receive result): 32 CPI (each)

c) What is the minimum number of points tested that will justify 2-way parallelization? (based on your algorithm's pseudo-code)

Q2. [Amdahl's Law] 20 pts

You have two applications running on a dual-core CPU. Their resource requirements are not equal - the first application needs 75% of the resources, and the other only 25% of the resources.

a) Given that 60% of the first application is parallelizable, how much speedup would you achieve with that application if run in isolation?

b) Given that 95% of the second application is parallelizable, how much speedup would this application observe if run in isolation?

c) Given that 60% of the first application is parallelizable, how much *overall system speedup* would you observe if you parallelized it, but not the second application?

d) How much overall system speedup would you achieve if you parallelized both applications, given the information in parts (a) and (b)?