

Homework 3

Files to submit: **pi.c**

Time it took Matthew to Complete: **20 mins**

- All programs must compile without warnings when using the -Wall and -Werror options
- Submit only the files requested
 - Do **NOT** submit folders or compressed files such as .zip, .rar, .tar, .targz, etc
- Your program must match the output exactly to receive credit.
 - Make sure that all prompts and output match mine exactly.
 - Easiest way to do this is to copy and paste them
- All input will be valid unless stated otherwise
- Print all real numbers to two decimal places unless otherwise stated
- The examples provided in the prompts do not represent all possible input you can receive.
- All inputs in the examples in the prompt are underlined
 - You don't have to make anything underlined it is just there to help you differentiate between what you are supposed to print and what is being given to your program
- If you have questions please post them on Piazza

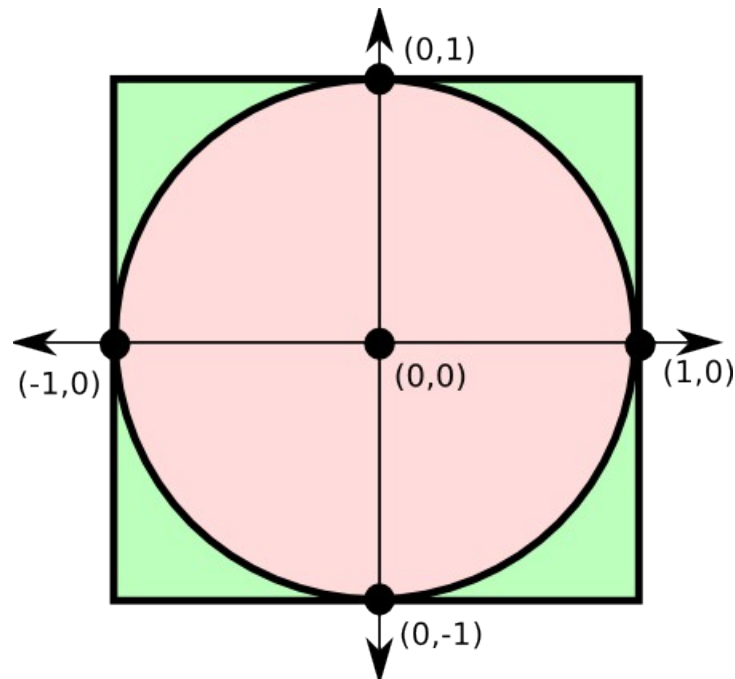
Restrictions

- No global variables are allowed
- Your main function may only declare variables, call other functions, and assign variables values.

Assumptions

- Input is **NOT** guaranteed to be valid
 - If invalid input is entered your program should continue to ask the user for input until valid input is entered

For this problem you will be implementing a Monte Carlo method for estimating the value of PI. Imagine that you have a circle of radius 1 whose center is at (0,0) surrounded by a square with sides length 2 (Pictured below).



If we randomly choose points within the square the probability of a point being within the circle is proportional to the area of the square it occupies. This means that

$$Probability_{Inside\ Circle} = \frac{Area_{Circle}}{Area_{Square}} = \frac{\pi * r^2}{2 * 2} = \frac{\pi * 1^2}{4} = \frac{\pi}{4}$$

Solving for pi we have $\pi = Probability_{Inside\ Circle} * 4$. The probability of being inside the circle can be approximated by randomly picking a large number of points between [-1,1] and counting the number that are inside the circle. The number of points that end up inside the circle divided by the total number of points picked is the probability of being inside the circle.

$$Probability_{Inside\ Circle} = \frac{number\ of\ points\ in\ the\ circle}{number\ of\ points\ picked}$$

Input Constraints

- Seed: an integer
- Number iterations: an integer greater than 0

Randomness

- In order to match the tester output you must make calls to rand in the exact order that I do
- First generate a random number from [-1, -1] for the X axis
- Then generate a random number from [-1, -1] for the Y axis
- You should use this expression for generating a random double between LO and HI
 - `LO + ((double) (rand())) / (((double) RAND_MAX) / (HI - LO));`
- Numbers that fall exactly on the edge of the circle are considered to be inside the circle

Examples

1. Enter the seed for the random number generator: 4
Enter the number of iterations to run: 10
The value of pi is 3.20000.
2. Enter the seed for the random number generator: -7
Enter the number of iterations to run: 2
The value of pi is 4.00000.
3. Enter the seed for the random number generator: -7
Enter the number of iterations to run: 10000
The value of pi is 3.14360.
4. Enter the seed for the random number generator: 3.45
Enter the seed for the random number generator: why
Enter the seed for the random number generator: 12
Enter the number of iterations to run: cat
Enter the number of iterations to run: -1
Enter the number of iterations to run: 3.141
Enter the number of iterations to run: 75
The value of pi is 3.14667.