Mathematics 1262 (C++ Programming) Hilary 2014

Fifth assignment, due 21/2/14

This assignment is about a 'Monte-Carlo method' of estimating π. Given a random point (x, y) in the unit square $0 \leq x, y \leq 1$, its probability of lying in the unit circle is $\pi/4$.

Take a sequence of n random points in the unit square, and count how many are in the unit circle: if c is the answer, then $4*c/n$ is an estimate of π. Be careful that this expression is calculated properly in double precision.

The only input to the program is the number n, which is read off the command line. Use `atoi()` to convert it.

Include a global boolean variable `seeded`, a routine `seed()`, and a function `double rand_double()`

```
#include<iostream>
#include<cstdlib>
#include<sys/time.h>
using namespace std;
bool seeded = false;
void seed()...
double rand_double()...
int main( int argc, char * argv[])...
```

`Rand_double()` produces double-precision numbers between 0 and 1. It must consult `seeded` to see whether `seed()` has been called; `seed()` should use the system clock. See the notes.

```
Sample output
% a.out 10
estimate of pi 2.4
% a.out 100
estimate of pi 3.08
% a.out 1000
estimate of pi 3.22
% a.out 1000000
estimate of pi 3.14279
% a.out 1000000
estimate of pi 3.14304
```

Motivation. Using random numbers.