Assignment 2 **159.240** **Due: Friday 19 December 2014**

Monte Carlo Pi Calculation

The task:

You must calculate Pi correct to 3 decimal places using Monte Carlo Simulation. You are allowed to use `rand()` and `srand()`.

Follow these steps:

1. Set up a new program
2. Make a loop that runs for N times (you choose N, make it 100 or a million)
3. Make two uniform random floats (between 0 and 1) inside the loop
4. Treat the two floats as a coordinate point, and calculate its magnitude using Pythagoras (hint, you need to do `sqrt(x*x+y*y)`. `sqrt()` is a function inside `math.h`)
5. Check if the magnitude is less than one. If it is, then it is in the circle.
6. Total the number of random points that are inside the circle (call this number M).
7. Once the loop is finished running, divide M by N, and multiply the result by 4.0f.
8. You should get Pi from this calculation.

Notes:

As with assignment 1, be careful with truncation errors, these happen when you convert from an int to a float, by doing something like this:

```
int a;
float b = 1.7;
a = b;
// a will be 1, not 2
```

If you need an extra challenge, replace the uniform random distribution with the exponential random distribution from assignment 1, and see what happens to the value of Pi. Note: this won't count for extra marks...

Marking:

The assignment is worth 10 marks, and these are awarded as follows:

- 4 marks: Pi correct to 3 decimal places.
- 6 marks: no formula errors (magnitude calculation), no truncation errors.

Submitting

Please submit your .cpp source file via Stream. You are welcome to develop your assignment on a home computer, but you **must** ensure that your program works in the labs.

If you have any questions about this assignment, please email the lecturer.

a.v.husselmann@massey.ac.nz