Parallel Programming Languages & Systems
Exercise Sheet 1

This exercise sheet is assessed and must be your own work. It accounts for 10% of the course final mark, with a further 10% coming from the second exercise sheet. The best marks will be awarded for simple, correct and clearly argued content. The deadline is **4pm, Thursday 16th February, 2017**. *You should submit your work electronically, as a pdf document using the DICE command*

```
submit ppls 1 pplsEx1.pdf
```

It is important that you use the filename as given, and that your document is in pdf format. Please be sure that you have read, understood and adhered to the School and University guidelines on late submission of coursework and academic misconduct, which can be found via the course webpage. Furthermore, you are required to take reasonable measures to protect your assessed work from unauthorised access. For example, if you put any such work on a public repository then you must set access permissions appropriately (generally permitting access only to yourself, or your group in the case of group practicals).

Consider the following program, assuming atomic reads and writes and a sequentially consistent model of shared memory, and that programs use an integer type which can store arbitrarily large positive or negative values. In other words, there is no issue about "wrapping around" when integer arithmetic would otherwise overflow.

```
int x = 10, y = 0;
co
  {
    while (x!=y) {
      x = x-1;
    }
    y = y+1;
  }
//
  {
    <await (x==y);>
    x = 8;
    y = 2;
  }
oc
```

Discuss the full range of possible outcomes of this program, including those which fail to terminate, indicating the final values of x and y where these are stable, or describing their behaviour otherwise.

(100 *marks*)