# Tutorial Problems
# Formal Verification Techniques for GPU Kernels

Alastair F. Donaldson

January 2013

1. Consider the following example GPU kernel:

```
__kernel void foo(__local int *A, __local int *B) {
  A[tid] = B[tid];
  B[tid/2] = A[tid + 1];
  barrier();
  A[tid] = A[tid + 1];
  B[tid] = B[A[tid + 2]];
}
```

   Suppose this kernel is executed by two threads. Identify all the possible data races that might arise during execution, assuming that `A`, `B` and `C` point to disjoint arrays and that no array bounds exceptions occur.

   Explain how temporary variables and extra barriers could be inserted to eliminate the data races, using as few barriers as possible.

2. Consider the following example GPU kernel:

```
__kernel void foo(__local int *A) {
  int temp;
  int i = 1;
  while(i < tid) {
    temp = A[tid - i];
    barrier();
    A[tid] = A[tid] + temp;
    barrier();
    i = i * 2;
  }
}
```

   This kernel suffers from *barrier divergence*. Explain what the problem is.

   The programmer had intended to write a kernel where on each loop iteration half of the threads stop executing the loop body. How could this kernel be re-written to achieve this correctly?

3. Consider this simple kernel:

```
__kernel void foo(__local int *A, __local int *B, __local int *
    C) {
  A[tid] = B[tid];
  B[tid + 1] = C[tid + 1];
  C[tid + 1] = A[tid];
}
```

   Write down the sequential program generated by the GPUVerify technique in order to verify this kernel. You do not need to present bodies for the `LOG`, `CHECK` or `barrier` procedures.

   Explain how any data race bugs are detected by GPUVerify's race checking instrumentation for this example.

4. Treating reads from the shared state abstractly can lead to GPUVerify falsely reporting a data race in a kernel that is actually correct. Write a simple kernel which illustrates this.

5. GPUVerify does not perform array bounds checking. It is possible that lack of bounds checking could lead to the tool reporting false negatives with respect to data race analysis. Explain, using an example, how this could happen.