isExecuted_goofup<<<2,3>>>(dev_a, blockid = 1, threadid = 1);

- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 0, threadIdx.x = 0, ... ); → *dev_a is set to 0.
- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 0, threadIdx.x = 1, ... ); → *dev_a is set to 0.
- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 0, threadIdx.x = 2, ... ); → *dev_a is set to 0.
- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 1, threadIdx.x = 0, ... ); → *dev_a is set to 0.
- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 1, threadIdx.x = 1, ... ); → *dev_a is set to 1.
- isExecuted_goofup(dev_a, blockid = 1, threadid = 1, blockIdx.x = 1, threadIdx.x = 2, ... ); → *dev_a is set to 0.

If block 1 thread 1 finishes last, then *dev_a = 1.
If block 1 thread 1 does not finish last, *dev_a = 0.

We can't know or control which thread finishes last!