Module 4.0

Imagine that we a zipping two tensors `a` and `b` of shape (I, 1) and (1, J) where I is very large and J is a small fixed size. (this is a special case of the zip in the homework)

```python
def specialized_zip_cuda(out, a, b):

    x = numba.cuda.blockIdx.x * THREADS + numba.cuda.threadIdx.x
    thread_x = numba.cuda.threadIdx.x

    shared = numba.cuda.shared.array(J)
    if x < I:

        if thread_x < J:

            shared[thread_x] = b[thread_x]

        numba.cuda.syncthreads()

        local = a[x]

        for j in range(J):

            out[x, j] = local * shared[j]
```

---

1 1 point

Imagine that I = 1000 and J = 100, what is the maximum number global *reads* are there from any thread?

Type your answer...

---

2 1 point

Imagine that I = 1000 and J = 100, how many shared *reads* are there from each thread?

Type your answer...

---

3 1 point

Imagine that I = 1000 and J = 100, what is the max number of shared memory *writes* from any thread?

Type your answer...