Pre-meeting

- Don't think can use map to achieve parallelisation of nearest neighbour since each element of the output array is based on it's indices rather than the value currently there.
- So current strategy is to split / join
- Are able to look at dimensions one at a time as won't affect results.

During the meeting

- Limit application to only upscale image to strict multiples of input image dimensions
- In this case try to think about it as a map
- (2x upscaling) Map over input pixels, and for each one produce a 2x2 outimage with the same values inside. Then combine them together in some way.
- Goes from 2D input to 4D intermediate to 2D output
- Be careful about swapping dimensions in the middle step

- Implement in scala first without any lift
- Key ingredients: array (function that creates 2x2 array from single value), transpose, map, join.

- Generate function (take ith position and return some value based on it) since above array function is not implemented yet.
- Be sure to type check so that we do in fact generate a 4D array for instance or just to see what i'm doing wrong.

For next week: Have some scala code that type checks that we can run.