Task 1.
Prove that:

$$(reduce\ myop\ e)\ .\ (map\ f)\ ==\ (reduce\ myop\ e)\ .\ (map((reduce(myop\ e))\ .\ (map\ f))\ .\ distr\_p$$

By applying the hint given in the assignment, I get the following:

$$(reduce\ myop\ e)\ .\ (map\ f)\ ==\ (reduce\ myop\ e)\ .\ (map\ f)\ .\ (reduce(++)[])\ .\ distr\_p$$

Then apply theorem 2 of list homomorphism given in the lectures on the right side of the equation:

$$(reduce\ myop\ e)\ .\ map\ f)\ ==\ (reduce\ myop\ e)\ .\ (reduce(++)[])\ .\ (map(map\ f))\ .\ distr\_p$$

Applying theorem 3 on the right side:

$$(reduce\ myop\ e)\ .\ map\ f)\ ==\ (reduce\ myop\ e)\ .\ (map((reduce(myop\ e))\ .\ (map(map\ f))\ .\ distr\_p$$

Applying theorem 1 on the right side:

$$(reduce\ myop\ e)\ .\ (map\ f)\ ==\ (reduce\ myop\ e)\ .\ (map((reduce(myop\ e))\ .\ (map\ f))\ .\ distr\_p$$

Task 3:

I've been unable to solve the task, as I'm having trouble segmenting the array, from there on I just need to apply parFilter on it. This could be done easily with a fold I believe, but I don't think that it would be running in parallel.

Task 4:
Sweet spot is at 700, where the GPU start to outperform the CPU. I believe it is because of the overhead where the GPU needs to use more computing time to setup for parallel than it takes for the CPU to finish. As it can be noted, the GPU has a constant finish time when performing on a low number of elements, as it finishes it on cycle.