

Assignment 1

Programming Massively Parallel Hardware

Mads Thoudahl /qmh332

September 7, 2015

1 Task 1 - Proof

$$redomap(\odot)fe_{\odot} \tag{1}$$

$$=(redomap(\odot)e_{\odot}).(mapf) \tag{2}$$

$$id = (reduce(++))\square\distr_p \tag{3}$$

$$\stackrel{LHI2}{=} (reduce(\odot)e_{\odot}).(reduce(++))\square.(map(mapf)).\distr_p \tag{4}$$

$$\stackrel{LHI3}{=} (reduce(\odot)e_{\odot}).((map)reduce(\odot)e_{\odot}).\distr_p \tag{5}$$

2 Task 2 - Longest Satisfying Segment Problem

The assignment had to be extended to work as supposed, as firstx always took the value 0 thereby making the neutral element nonneutral...

```
lssop (lssx, lisx, lcsx, tlx, firstx, lastx, okx)
(lssy, lisy, lcsy, tly, firsty, lasty, oky) =
  (newlss, newlis, newlcs, tlx+tly, firstx, lasty, newok)
  connect = if (tlx==0 || tly==0) then True else p [lastx,firsty]
  newlss   = lssx 'max' lssy 'max' (if connect then (lcsx + lisy) else 0)
  newlis   = if (okx && connect) then (tlx + lisy) else lisx
  newlcs   = if (oky && connect) then (tly + lcsx) else lcsy
  newok    = okx && oky && connect
  fx       = if (tlx==0) then firsty else firstx
  ly       = if (tly==0) then lastx else lasty
```

3 Task 3

4 Task 4 - CUDA exercise

The code is implemented in the attached zip file under the `cuda` directory as `ass1.cu` and with a makefile as described. It has some global variables like number of threads and whether to time memory transfers set in define clauses in the file.

4.1 timings

Table 1: timings - avg of 3 runs all on gpu01 at diku server

800.000 calcs	GPU	CPU
with MEM	3.11 ms	3.19 ms
without MEM	0.165 ms	3.26 ms

All calculations are within a fault tolerance (EPSILON) of 0.0005. It seems that with the current calculations, around 14200 calculations make it break even. But if you do count the memory transfers in, it barely break even at 800.000 calculations. There is a define statement in the code to switch timing of memory transfers.