Lab 7_1 Documentation

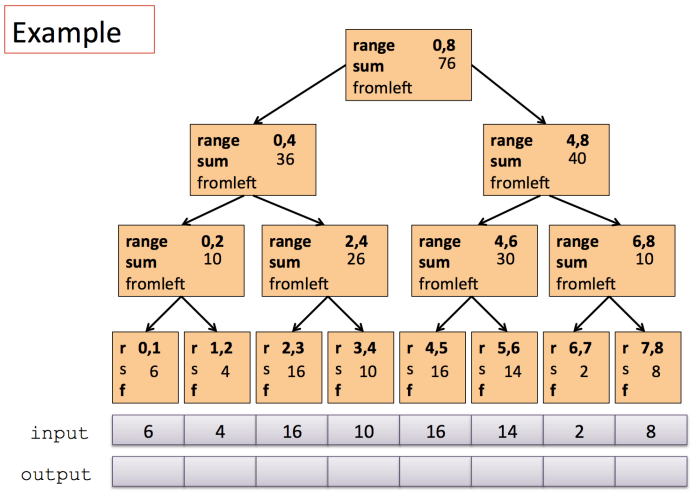
Requirement

1. Given a sequence of *n* numbers, compute the sums of the first *k* numbers, for each *k* between 1 and *n*. Parallelise the computations, to optimise for low latency on a large number of processors. Use at most 2**n* additions, but no more than 2*log(*n*) additions on each computation path from inputs to an output. Example: if the input sequence is 1 5 2 4, then the output should be 1 6 8 12.

In order to solve the requirement i used 2 passes:

- first pass: builds a tree of sums bottom-up (upPass)

  - build a binary tree where each node has a Sum and a fromLeftSum, as you go up the tree you add the Sum of the leafs to get the next node sum.



- second pass: we pass the fromLeftSum from the root the the leafs as following:

  - root has a fromLeftSum 0 and it passes to its left child the same fromLeftSum and to its right child, its fromLeftSum plus it's left child Sum and so on until we reach the leaves.