Sorting Networks

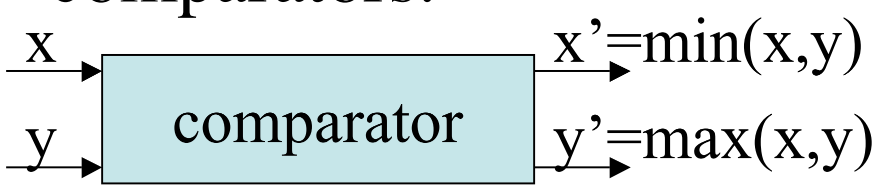
CS255
Chris Pollett
Feb. 8, 2006.

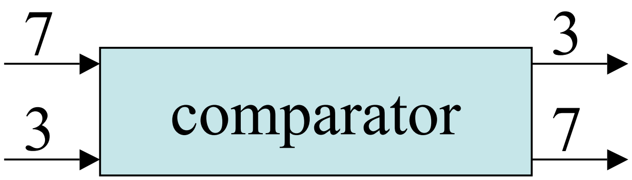
Outline

- Comparison networks
- The zero-one principle
- A bitonic sorting network

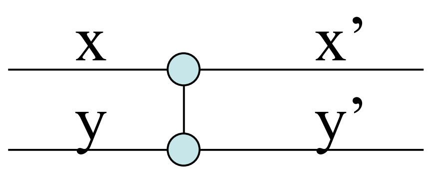
Comparison networks

- Today, we look at the first of three parallel computation models we'll consider this semester: Sorting networks.
- The basic building blocks of our networks will be comparators:





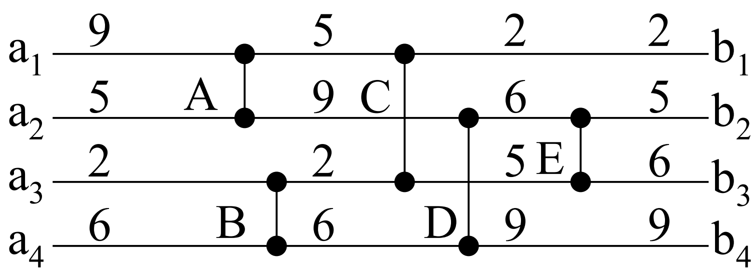
- We'll often draw such comparators more simply as:



Some Definitions

- A **wire** transmits a value from place to place. A wire may connect to a comparator and a wire may lead from a comparator.
- Wire can have labels like a_i, or x_i.
- We'll assume our networks consists of n **input wires** $a_1,\ldots,a_n$. These wires don't connect from any comparators, but instead connect from vertices consisting of an **input sequence** of values which we'll also write as $a_1,\ldots,a_{n.}$
- We'll also assume our networks consist of n **output wires** $b_1,\ldots,b_n$. These wires don't connect to any comparators, but instead connect to a set of vertices which will receive an **output sequence** of values which we'll also write as $b_1,\ldots,b_n$. This sequence is supposed to be the result of performing the comparisons in the networks to the input sequence.

Example Comparison Network



A,B are depth 1

C,D are depth 2

E is depth 3

- We require that our networks be acyclic.
- The **depth** of an input wire is 0. The **depth** of a wire leading out of a comparator is 1 plus the depth of its inputs. The **depth** of a comparison network or a comparator is the maximum depth of its inputs.
- In computing the values of the wire in a network one start at the values of the wires of depth 0 (the inputs), then all the values of depth 1 are computed simultaneously using the values of the depth 0 wires and the relevant comparators., then those of depth 2, etc.
- A **sorting network** is a comparison network for which the output sequence is nondecreasing. i.e., $b_1 \le b_2 \le \ldots b_n$.

The zero-one principle

- The zero-one principle says that if a sorting network works correctly when each input is drawn from the set {0, 1}, then it works on arbitrary numbers.
- We'll now try to argue this principle is true.

Lemma 27.1

Lemma: If a comparison network transforms the input sequence a=<a_1,..,a_n> into the output sequence b=< b_1,..,b_n >, then for any nondecreasing function f, the network transforms the input sequence $f(a)$=<$f(a_1)$,..,$f(a_n)$> into $f(b)$=<$f(b_1)$,..,$f(b_n)$>.

Proof: We first prove the claim for a single comparator and then prove the result by induction. Since f is nondecreasing x<= y implies $f(x)$<= $f(y)$, so min($f(x)$, $f(y)$) = f(min(x,y)). Similarly, max($f(x)$, $f(y)$) = f(max(x,y)). So this proves the claim for a comparator. We now prove it true for a network by induction on its depth. When the depth is 0 we have no comparators in our network and the result is trivial. Now assume that for all networks of depth strictly less than d >=1, that if an output wire assumes the value a_i when the input a is applied, then if assume the value $f(a_i)$ when $f(a)$ is applied. Suppose we have an output wire a_k of depth d in a network of depth d. It must be the result of a comparator whose inputs must be wires a_i and a_j of depth strictly less than d. i.e., it must be either min(a_i, a_j) or max(a_i, a_j). By the induction hypothesis, we know on input $f(a)$ these wires carry value $f(a_i)$ and $f(a_j)$. By our result about comparators, the outputs of the comparator on these two values is f(min(a_i, a_j)) and f(max(a_i, a_j)). So the induction holds. q.e.d.

Bitonic Sequences

- In order to develop our sorting networks we will first consider sorters that can sort bitonic sequences.
- A **bitonic sequence** is a sequence which consists of a nondecreasing sequence followed by a nonincreasing sequence. We also call cyclic shifts of bitonic sequence, bitonic sequences.
- For example, <1,1,2,3,3,5,2>, <0, 0, 1,1,1,1,0>, <6,9,4,2,3,5> (The last is an example with a cyclic shift of 3).
- Zero-one bitonic sequences have the form $0^i1^j0^k$ or $1^i0^j1^k$ for nonnegative i,j,k.