$\Omega(n \log n)$ element comparisons in the worst case.

12.5 Linear Time Reductions

For the problem ELEMENT UNIQUENESS, we were able to obtain a lower bound using the algebraic decision tree model of computation directly by investigating the problem and applying Theorem 12.4. Another approach for establishing lower bounds is by the use of reductions. Let A be a problem whose lower bound is known to be $\Omega(f(n))$, where $n = o(f(n))$, e.g. $f(n) = n \log n$. Let B be a problem for which we wish to establish a lower bound of $\Omega(f(n))$. We establish this lower bound for problem B as follows.

(1) Convert the input to A into a suitable input to problem B.
(2) Solve problem B.
(3) Convert the output into a correct solution to problem A.

In order to achieve a linear time reduction, Steps 1 and 3 above must be performed in time $O(n)$. In this case, we say that the problem A has been reduced to the problem B in linear time, and we denote this by writing

$$A \propto_n B.$$

Now we give examples of establishing an $\Omega(n \log n)$ lower bound for three problems using the linear time reduction technique.

12.5.1 *The convex hull problem*

Let $\{x_1, x_2, \ldots, x_n\}$ be a set of positive real numbers. We show that we can use *any* algorithm for the CONVEX HULL problem to sort these numbers using only $O(n)$ time for converting the input and output. Since the SORTING problem is $\Omega(n \log n)$, it follows that the CONVEX HULL problem is $\Omega(n \log n)$ as well; otherwise we will be able to sort in $o(n \log n)$ time, contradicting Theorem 12.5.

With each real number x_j, we associate a point (x_j, x_j^2) in the two dimensional plane. Thus, all the n constructed points lie on the parabola $y = x^2$ (see Fig. 12.2).

If we use any algorithm for the CONVEX HULL problem to solve the constructed instance, the output will be a list of the constructed points sorted by their x-coordinates. To obtain the sorted numbers, we traverse

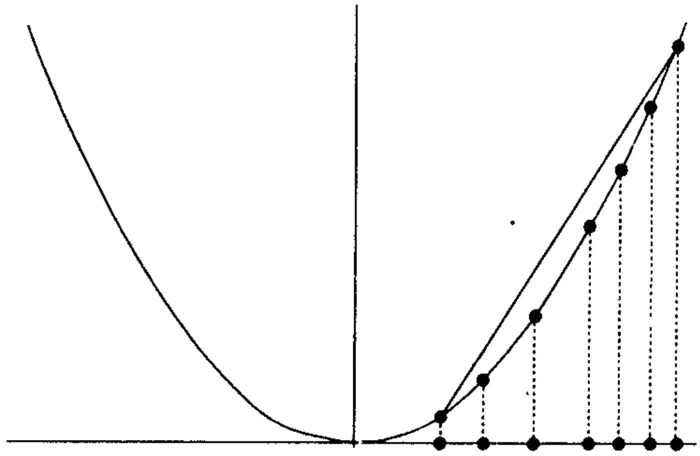

Fig. 12.2 Reducing sorting to the CONVEX HULL problem.

the list and read off the first coordinate of each point. The result is the original set of numbers in sorted order. Thus we have shown that

$$\text{SORTING} \propto_n \text{CONVEX HULL},$$

which proves the following theorem:

Theorem 12.7 In the algebraic decision tree model of computation, any algorithm that solves the CONVEX HULL problem requires $\Omega(n \log n)$ operations in the worst case.

12.5.2 *The closest pair problem*

Given a set S of n points in the plane, the CLOSEST PAIR problem calls for identifying a pair of points in S with minimum separation (see Sec. 6.9). We show here that this problem requires $\Omega(n \log n)$ operations in the worst case by reducing the problem ELEMENT UNIQUENESS to it.

Let $\{x_1, x_2, \ldots, x_n\}$ be a set of positive real numbers. We show that we can use an algorithm for the CLOSEST PAIR problem to decide whether there are two numbers that are equal. Corresponding to each number x_j, we construct a point $p_j = (x_j, 0)$. Thus the constructed set of points are all on the line $y = 0$. Let A be any algorithm that solves the CLOSEST PAIR problem. Let $(x_i, 0)$ and $(x_j, 0)$ be the output of algorithm A when presented with the set of constructed points. Clearly, there are two equal numbers in the original instance of the problem ELEMENT UNIQUENESS if