

of algorithms. Algorithms are at the core of most technologies used in contemporary computers.

Furthermore, with the ever-increasing capacities of computers, we use them to solve larger problems than ever before. As we saw in the above comparison between insertion sort and merge sort, it is at larger problem sizes that the differences in efficiency between algorithms become particularly prominent.

Having a solid base of algorithmic knowledge and technique is one characteristic that separates the truly skilled programmers from the novices. With modern computing technology, you can accomplish some tasks without knowing much about algorithms, but with a good background in algorithms, you can do much, much more.

## Exercises

### 1.2-1

Give an example of an application that requires algorithmic content at the application level, and discuss the function of the algorithms involved.

### 1.2-2

Suppose we are comparing implementations of insertion sort and merge sort on the same machine. For inputs of size  $n$ , insertion sort runs in  $8n^2$  steps, while merge sort runs in  $64n \lg n$  steps. For which values of  $n$  does insertion sort beat merge sort?

### 1.2-3

What is the smallest value of  $n$  such that an algorithm whose running time is  $100n^2$  runs faster than an algorithm whose running time is  $2^n$  on the same machine?

---

## Problems

### 1-1 *Comparison of running times*

For each function  $f(n)$  and time  $t$  in the following table, determine the largest size  $n$  of a problem that can be solved in time  $t$ , assuming that the algorithm to solve the problem takes  $f(n)$  microseconds.

	1 second	1 minute	1 hour	1 day	1 month	1 year	1 century
$\lg n$							
$\sqrt{n}$							
$n$							
$n \lg n$							
$n^2$							
$n^3$							
$2^n$							
$n!$							

---

## Chapter notes

There are many excellent texts on the general topic of algorithms, including those by Aho, Hopcroft, and Ullman [5, 6]; Baase and Van Gelder [28]; Brassard and Bratley [54]; Dasgupta, Papadimitriou, and Vazirani [82]; Goodrich and Tamassia [148]; Hofri [175]; Horowitz, Sahni, and Rajasekaran [181]; Johnsonbaugh and Schaefer [193]; Kingston [205]; Kleinberg and Tardos [208]; Knuth [209, 210, 211]; Kozen [220]; Levitin [235]; Manber [242]; Mehlhorn [249, 250, 251]; Purdom and Brown [287]; Reingold, Nievergelt, and Deo [293]; Sedgewick [306]; Sedgewick and Flajolet [307]; Skiena [318]; and Wilf [356]. Some of the more practical aspects of algorithm design are discussed by Bentley [42, 43] and Gonnet [145]. Surveys of the field of algorithms can also be found in the *Handbook of Theoretical Computer Science, Volume A* [342] and the *CRC Algorithms and Theory of Computation Handbook* [25]. Overviews of the algorithms used in computational biology can be found in textbooks by Gusfield [156], Pevzner [275], Setubal and Meidanis [310], and Waterman [350].