



Overview

We complete our study of string processing by considering a number of ingenious algorithms that take full advantage of several of the fundamental methods covered earlier in the course. In the first lecture, we consider regular expression pattern matching, where the goal is to find strings from a specified set in a given text. In the second lecture, we consider classic methods for data compression.

Lecture 9: Regular Expressions. A regular expression is a method for specifying a set of strings. Our topic for this lecture is the famous grep algorithm that determines whether a given text contains any substring from the set. We examine an efficient implementation that makes use of our digraph reachability implementation from Week 1.

Lecture 10: Data Compression. We study and implement several classic data compression schemes, including run-length coding, Huffman compression, and LZW compression. We develop efficient implementations from first principles using a Java library for manipulating binary data that we developed for this purpose, based on priority queue and symbol table implementations from earlier lectures.

To Do:

- **Job Interview Questions.** Algorithmic interview questions based on the lecture material.
- **Programming Assignment: Burrows-Wheeler.** Implement the Burrows-Wheeler data compression algorithm.
- **Suggested Readings.** Section 5.4 and 5.5 in *Algorithms, 4th edition*.

✓ Complete

Go to next item

