Assignment 3

(Revised Version of October 15)

CSE 326: Data Structures and Algorithms

Autumn Quarter, 1999, University of Washington.

Steven Tanimoto, Instructor

Due dates: For Part 1, October 20; for Part 2, October 27.

Part 1 (Due Wednesday, October 20, in class)

1. Problem 1 on page 90 of Lewis and Denenberg.
2. Problem 1 on page 126.
3. Problem 6 on page 126.
4. Problem 12 on page 127.
5. Problem 14 on page 127.
6. (optional[1]) Problem 28 on page 129.

Part 2 (Due Wednesday, October 27, at 5:00)

For this problem, you are encouraged to work in teams of two people. In most cases, both partners in a team will receive the same grade.

Write and test a program that reads a text file, determines the frequency of occurrence of each character occuring in the file, and builds a Huffman tree for that set of character frequencies. Then it encodes the file as a bitstream with the description of the code at the beginning of the file. Write and test a companion program that decodes the compressed file. Determine your compression rates on three example files – a small one (about 100 bytes), a medium-sized on (about 5K bytes) and a large one (about 50 K bytes).

(optional) Implement the adaptive Huffman encoding method described on pages 150, problem 31 on pages 169-170 and 484, which avoids having to transmit a description of the code.

In order to simplify the coding, we plan to post example code for the reading and writing of binary data (which is typically how one accesses files at the bit level).

Alternatively, you may choose to output bits using one full character (byte) per bit, and then assume that your files are 1/8 as large as the "ls" command says they are.

---

[1] On this assignment, an optional problem is worth points, but not very many points when considered from a points/effort standpoint. Thus an optional problem is intended to offer an additional challenge, but one which will not substantially penalize anyone who doesn't solve it.