Advanced Topics of Algorithms

Tasks 1, 19.1.2016

Do at least <u>three</u> of the basic tasks, and select <u>one</u> of the challenging tasks (group work allowed).

BASIC TASKS

1. We have an alphabet of five symbols {A,I,K,N,U} and three different code tables defined in the table below. Draw the corresponding code trees for these code tables. Are these code tables (a) prefix codes, (b) uniquely decodable? Decompress the following (compressed) bit stream using these three code tables: 0110011101101110011000.

Symbol	Code 1	Code 2	Code 3
A	0	00	0
I	100	01	010
K	101	11	1100
N	110	10	1101
U	111	110	111

2. You have to code the observations from 100 days whether you saw a dog in streets or not. Before coding, you must decide the probability model you are going to use for it. Consider the following models for seeing a dog: (a) 75%, (b) 50%, (c) 25%, (d) 10%, (e) 0%. Suppose then that you will see a dog every 3rd day, on average, and that the probability of seeing a dog is independent of the previous days. Calculate the entropies and the overall bits needed using the models (a)-(e). Which model is the best? Is it optimal? If not, what would be the optimal model?

3. We have an input alphabet consisting of the symbols {a, b, c}, and their probabilities {0.7, 0.2, 0.1} respectively. Calculate their entropy. (a) What is the code length when coding sequence (a, a, b, a, c) using these entropies? (b) What is the redundancy of this code? (c) Generate Huffman code for the same model and calculate the code length. (d) Explain the result.

4. Calculate context model for the binary image below using 1-pixel content model with: (a) pixel above, (b) pixel to the left. Which model provides smaller entropy? What is the entropy
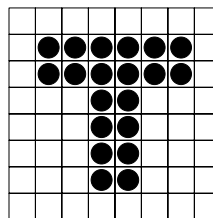
CHALLENGING TASKS

1. Make program that reads a web page, extracts all text from it (PHP component existing in home page). Output the statistics as text output or simple web page.

2. Calculates the frequencies of all letters 'a' to 'ö' and sort them in the decreasing order according to the frequencies.

3. Calculate entropies for each character. Calculate the total entropy.

4. Make program that creates Huffman code for the above. Plot the code.

5. Calculate the difference of the code and the entropy.

6. Test with several few random pages and report the average entropy and Huffman code length for these pages.