TDT4117 Information Gathering

Assignment 5

Erling Storaker Moen

Task 1-1: Text Compression with Huffman code

J=39 J = 100111 Q = 101

Q=5 Q = 101 Z = 1010

W=22 W = 10110 X = 1100

X=12 X = 1100 Æ = 10010

Y=31 Y = 11111 W = 10110

Z=10 Z = 1010 C = 11011

B=41 B = 101001 Y = 11111

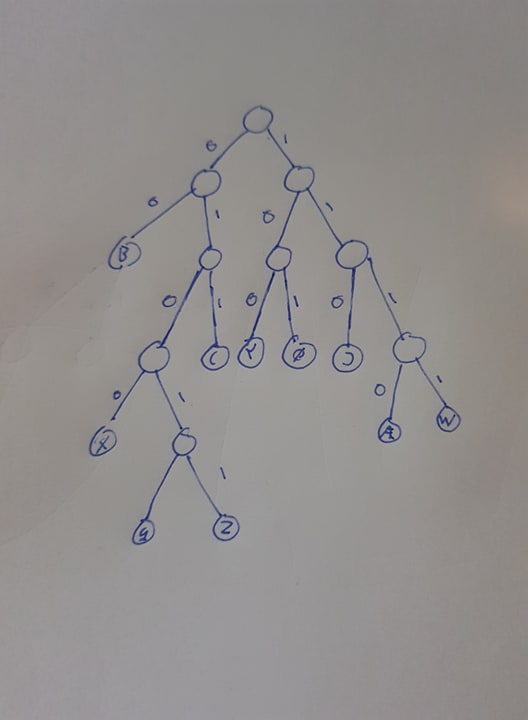
Æ=18 Æ = 10010 Ø = 100011

C=27 C = 11011 J = 100111

Ø=35 Ø = 100011 B = 101001

Huffman Tree

(‘Q’,3,101), (‘Z’,4,1010), (‘X’,4,1100), (‘Æ’,5,10010), (‘W’,5,10110), (‘C’,5,11011), (‘Y’,5,11111), (‘Ø’,6,100011), (‘J’,6,100111), (‘B’,6,101001),



Average length of code: (3+4+4+5+5+5+5+6+6+6)/10 = 4.9

Task 1-2:

Input :1011101011111111000110000010001111001

Starting from the left side, look for a unique length of bits that fit the code for a letter

Start: 1011101011111111000110000010001111001

1011101011111111000110000010001111001 = H

Remove 1011 from string

101011111111000110000010001111001 = E

Remove 1010 from string

11111111000110000010001111001 = L

Remove 1111 from string

1111000110000010001111001 = L

Remove 1111 from string

000110000010001111001 = O

Remove 000 from string

110000010001111001 = W

Starting to look like “Hello World..”

Remove 1100 from string

00010001111001 = O

Remove 000 from string

10001111001 = R

Remove 1000 from string

1111001= L

Remove 1111 from string

0011 = D

Remove 001

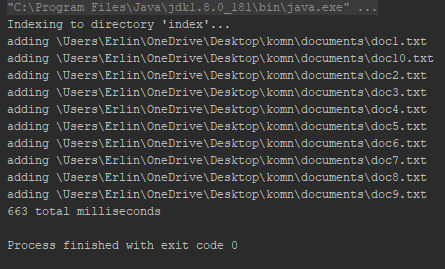
Solution = “Hello World”

Task 2: Index Analysis Using Lucene

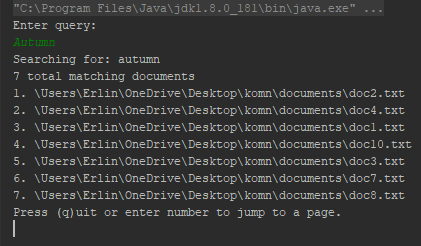
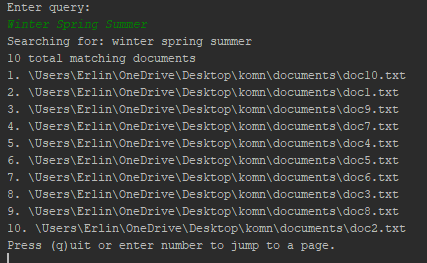
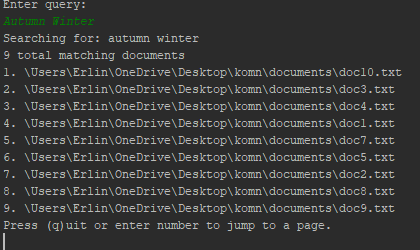
1. Lucene is a text search engine library. Without going too much into detail it is a technology that is suitable for nearly any application that requires text search. It has many advantages, like high speed, ranked searching, many query types, fielded searching and so on. It also works cross platform with Java.

I didn’t get the program to work with Lucene 7.5.0 because “LongField” was outdated, I went back a few versions and found Lucene 5.5 to work fine, therefore I used it instead.

2.



1. Autumn, Autumn Winter and Winter Spring Summer queries



I believe Lucene uses a vector space query model combined with the Boolean model for this query.

1. Query for NTNU returned one result: maildir\lay-k\inbox\1126

Time used: 11ms

Query for “Fireworks” surprisingly returned 140 results

Time used: 6ms

Query for “League of Legends” returned 2237 results, this may be because of the word “of”

Time used 23ms

Query for “Erling” returned 17 results

Time used:15ms

Query for “What does the fox say woof woof woof” returned 123158 results

Time used: 55ms