Computational Linguistics I, Winter 2006. Marcus Kracht

To be submitted: Friday, March 3, 2006. In these exercises we shall do a simple encoding of a file.

[A 6.1] Assume that the characters are given numbers. Counting the digits, the lower and upper case letters, period, comma and the blank, we use, let us say, exactly 65 characters. Give each character a number from 0 to 64. Write a function that shifts each character a given number of places forward in this table. Now write a function that "adds" two characters via the table as follows: if `b` has code 1 and `C` code 28, then the sum is 29, and this is the letter `D`.

[A 6.2] Now write a function that takes two strings as input. The first is the "codeword" and the second the string to be encoded. Use the codeword to clock forward the encoding in the following way: if the code word has length n, then the characters from the string to be encoded of places $n + i$, $n + 2i$, $n + 3i$ and so on are all moved forward as many places as the ith character of the codeword names, using the addition of the previous exercise. (So, we 'write' the codeword repeatedly under the string and add the characters vertically to get the result.) *Hint.* For the next assignment it is profitable to have an extra argument for the "offset". The offset will be the position on the codeword at which encoding starts.

[A 6.3] Write a function `encode codeword file` that does the following: it inputs data line by line from "file", performs the encoding of the previous exercise, treating the input as one continuous string, but ignoring linebreaks, and writes it out to the file "file"`.ncr`.