Do ONE of the following two exercises

**Exercise A**

Write a program that for a given bit vector $v$ builds a data structure such that subsequent bit-rank queries of the following kind can be answered in constant time: given $i$ with $1 \leq i \leq |v|$, compute $\text{rank}(i) = |\{j \leq i : v[j] = 1\}|$, that is, the number of 1-bits up to position $i$.

Try to choose the blocks and superblocks such that the data structure uses as little space (in addition to the $n$ bits used for the bit vector) as possible.

Check the correctness of your program by verifying that $\text{rank}(1) = v[1]$ and for every $i$ with $2 \leq i \leq |v|$, $\text{rank}(i) - \text{rank}(i - 1) = v[i]$.

Run your program with the text from Exercise Sheet 1 (with each character contributing 8 bits in the obvious way) as input, and post the size of this bit vector and the additional number of bits used by your data structure on the Wiki.

**Exercise B**

Write a program that for a given string computes the Burrows-Wheeler transform (the $B$ array from our lecture, which is easily obtained from the suffix array, which, in turn, you can compute with your program from Exercise Sheet 1), then applies Move-To-Front (MTF) encoding, and encodes the resulting sequence of integers with either Elias-$\gamma$ or Elisa-$\delta$ encoding (your choice). The result is a bit vector.

Check the correctness of your program on a few special strings, like $abab\ldots ab$, for which you can easily predict the correct result.

Run your program with the text from Exercise Sheet 1 as input, and post the size of the string and the size of the resulting bit vector on the Wiki.