4. How can you build a small index for a text S_3 such that later, given a pattern P, in $\mathcal{O}(|P|)$ time you can find the leftmost occurrence of P in S_3? How can you modify it such that, given k, in $\mathcal{O}(|P| + \log |S|)$ time you can find the kth occurrence of P in S_3, counting from the left?

   Hint: Use a suffix tree and a range-minimum data structure for the first part of the question, then replace the range-minimum data structure by a wavelet tree for the second part.

5. Let S_4 and S_5 be two strings and let C be a longest common subsequence of $\mathsf{BWT}(S_4)$ and $\mathsf{BWT}(S_5)$. If S_4 and S_5 are very similar then C is usually nearly as long as they (and their BWTs) are. For example, if

$$S_4 = \mathsf{GCACTTAGAGGTCAGT} \qquad S_5 = \mathsf{GCACTAGACGTCAGT}$$

   then

$$\mathsf{BWT}(S_4) = \mathsf{TCTGCGTAAAAGGTGC} \qquad \mathsf{BWT}(S_5) = \mathsf{TGCTCGTAAAACGCG}$$

   and we can choose

$$C = \mathsf{TCTCGTAAAAGG}\,.$$

   Let D_1 and D_2 be the complements of C in $\mathsf{BWT}(S_4)$ and $\mathsf{BWT}(S_5)$, respectively, and let B_1 and B_2 be bitvectors with 1s marking the characters of D_1 and D_2 in $\mathsf{BWT}(S_4)$ and $\mathsf{BWT}(S_5)$. In our example

$$D_1 = \mathsf{GTGC} \qquad D_2 = \mathsf{GCC}$$

   and

$$B_1 = 0001000000000111 \qquad B_2 = 010000000001010\,.$$

   Suppose we have data structures supporting fast rank queries over $\mathsf{BWT}(S_4)$, D_1, D_2, B_1 and B_2, and fast select_0 queries over B_1. Explain

   - how we can use these data structures to support fast rank queries over $\mathsf{BWT}(S_5)$,
   - why this might be useful if we already have an FM-index for S_4 and we want to build a small FM-index for S_5.

   Hint: Remember that $\mathsf{BWT}(S_5).\mathsf{rank}_X(i)$ returns the number of Xs in $\mathsf{BWT}(S_5)[1..i]$, some of which are in D_2 and the rest of which are in C. To count the Xs in both $\mathsf{BWT}(S_5)[1..i]$ and C, find a position j such that the number of characters in both $\mathsf{BWT}(S_4)[1..j]$ and C is the same as the number in both $\mathsf{BWT}(S_5)[1..i]$ and C. The number of Xs in both $\mathsf{BWT}(S_4)[1..j]$ and C is the number in $\mathsf{BWT}(S_4)[1..j]$ minus the number in both $\mathsf{BWT}(S_4)[1..j]$ and D_1.