Exercise 1:
Compute the table $H$ for the pattern string 100011011011001.

Exercise 2:

a) Prove the correctness of the algorithm COMPUTATION OF $H$.

b) Give an exact analysis of the number of comparisons performed by the algorithm KMP.

c) For $1 \leq r \leq m$ we have computed the values $H(r) = \max_{l<r}\{l \mid b_1b_2\ldots b_{l-1} \text{ is suffix of } b_1b_2\ldots b_{r-1}\}$. It is obvious that in the case $b_{H(i)} = b_r$ there holds $b_{H(i)} \neq a_j$. Hence, instead of $H(r)$ it would be better to compute the value $\text{Next}(r)$ where $\text{Next}(r) = \max_{l<r}\{l \mid b_1b_2\ldots b_{l-1} \text{ is suffix of } b_1b_2\ldots b_{r-1} \text{ and } b_l \neq b_r\}$. Develop an efficient algorithm for the computation of the table $\text{Next}$. Prove the correctness of your algorithm and analyze its needed time. Modify the algorithm KMP such that the table $\text{Next}$ instead of the table $H$ is used.

Exercise 3:
Modify the algorithm KMP such that all occurrences of $y$ in $x$ are computed in $O(n+m)$ time. Extend the algorithm COMPUTATION OF $H$ such that $H(m+1)$ is also computed.

Exercise 4:
Modify the algorithm KMP such that the longest prefix of $y$ which is a substring of $x$ is computed.