Exercises about Turing machines

1. Simulate the behaviour of a given Turing machine as a game. (Your exercise teacher will have some transition diagrams of different size of Turing machines.) If there are enough students in the group, you can make a competition between machines in recognizing a language. Each student corresponds a state, performs a required writing operation on the tape, moves the reading head and gives the control into the correct successor state. The accepting or rejecting final state reports the result. (One point for each participant of the game!)

2. Construct a standard type Turing machine, which recognizes the language $\{ww^R | w \in \{a,b\}^*\}$.

3. Construct a standard type Turing machine, which transform the string w into string ww^R ($w \in \{a,b\}^*$).

4. Construct a standard type Turing machine, which recognizes the language $\{w \in \{a,b\}^* | w$ contains equal number of a's and b's$\}$. Notice! a's and b's can appear in any order. E.g. *abab* and *bbaaba* belong to the language.

5. Construct a standard Turing machine, which subtracts one from the input binary string. I.e. an integer n is given as a binary string x, in which the most significant bits are left and least significant right. If $n > 0$, the machine replaces x by the binary representation of integer $n - 1$. If $n = 0$, the tape remains same.

6. Let's consider the following nondeterministic Turing machine:

   $M = (\{q_0, q_1, q_2, q_f\}, \{0,1\}, \{0,1\}, \delta, q_0, q_f, q_{no})$,

   whose transition diagram is defined as

   $\delta(q_0, 0) = \{q_0, 1, R), (q_1, 1, R)\}$
   $\delta(q_1, 1) = \{q_2, 0, L)\}$
   $\delta(q_2, 1) = \{q_0, 1, R)\}$
   $\delta(q_1, <) = \{q_f, <, R)\}$

   What does the machine do? Hint: simulate its behaviour with different binary strings. (You can use JFLAP, if you want.)

7. Construct a nondeterministic Turing machine, which recognizes the language $\{ww|w \in \{a,b\}^*\}$.

8. Consider the nondeterministic Turing machine TEST_COMPOSITE (look at the English material give to you), which recognizes composite numbers. Could you make a prime tester machine by changing tha accepting and rejecting states of that machine? Justify your answer!

9. Describe (unformally) a nondeterministic Turing machine, which recognizes the following language: The words of the language are of form $w_1\#w_2\#...\#w_n$ for any n such that for all i $w_i \in \{a,b\}^*$ and for some j w_j is the binary representation of integer j. N.B.! Utilize the nondeterminism as much as possible, i.e. prefer much branched but short paths. (The machine guesses the correct path nondeterministically.)