Unless stated otherwise, we assume that the original content of the tape is a finite sequence of 0's and 1's and the rest of it is filled with blank symbols. The original position of the Turing machine is assumed to be on the leftmost character of the sequence.

Exercise 1. Draw the diagram of a Turing machine that ends in the accept state if there are two consecutive 1's on the tape and ends in the reject state otherwise.

Exercise 2. Draw the diagram of a Turing machine that answers the question: are there more 1's than 0's on the tape ?

Exercise 3. Draw the diagram of a Turing machine that adds 1 to the binary number written on the tape. For this exercise we will assume that we start on the rightmost position of the sequence.

Exercise 4. Describe a Turing machine that given a finite number of x's on its initial tape returns the binary representation of the length of the original sequence.

Exercise 5 (busy beaver)**.** Let us consider an n state (without the halting one) Turing machine with only two output symbols 0 and 1. We suppose that such a machine starts on an all 0 tape and we want to know for a given n what is the maximum number of 1's that it can write on the tape given that it still halt at some point. This function, usually denoted $\Sigma(n)$, is known as the busy beaver function, and any Turing machine outputting this maximal number of 1's is known as a busy beaver.

For $n = 1$ for instance, if we do not want the machine to loop forever we can only output one 1 and go to the halting state. Hence $\Sigma(1) = 1$.

- Given that $\Sigma(2) = 4$ and $\Sigma(3) = 6$ can you find a busy beaver for $n = 2$ and $n = 3$?

Except for small n, this question is extremely difficult to answer. We know that $\Sigma(4) = 13$ and we don't know the value of Σ for any other n. What we know are only lower bounds, for instance $\Sigma(5) \geq 4098$ and $\Sigma(6) \geq 10^{1439}$.

As you can notice this function is increasing extremely quickly. Actually it is increasing more than any computable function, and it is itself a non-computable function (it is possible to define these notions precisely).