What is a TURING MACHINE?

A Turing machine is a computing device, just like your computer. You can think of it as a mechanism in which you write a word as input, and it processes it and outputs **yes** or **no** depending on whether or not such word satisfies a certain property. To understand this model, imagine an infinite tape divided in cells. Each cell either contains a symbol or is blank, and there is a read-write head pointing to exactly one cell at a time.
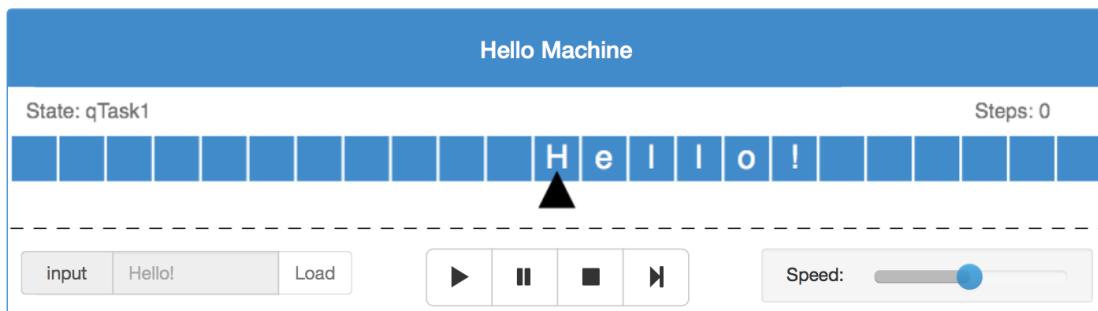

Figure 1

Besides the tape and the read-right head, a Turing Machine has a set of possible states. For the sake of simplicity, imagine that each state represents a task. Thus, the state of the machine represents the task the machine is working on.

When a Turing Machine is running, it is in exactly one state at a time. Every Turing Machine has an **initial state**. This state is adopted by the machine when it starts processing a word. At this time, the read-write head is pointing at the first symbol of the word.

The word processing is determined by the machine instructions, which are called transitions. Every transition has an executing condition and an Instruction. Here is an example of how a transition looks like.

Executing condition

> If the machine is in state qTask1 and the head is reading the letter 'H'

Instruction

> Switch to state qTask2, write a 'Y' and move the head right
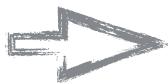
The executing condition depends only on the state and on what is being read. When the condition of a transition is satisfied, the instruction of the transition is executed.

Suppose that at some point of an execution, a certain machine looks like Figure 1. Thus, it is in state aTask1 and the head is reading the letter 'H'. At that point, the machine would satisfy the executing condition of the example transition. Hence, the transition's instruction would be executed. Accordingly, after applying this transition the machine would look like this:
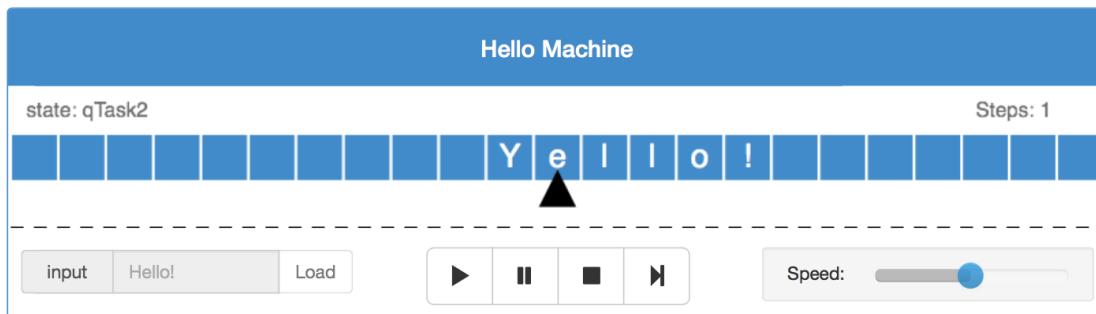

Figure 2

It is important to notice that the symbol is written <u>before</u> the head is moved. After this, the machine would continue running only if there was a rule with this condition:

The execution stops when the configuration of the machine doesn't satisfy any executing condition. Once the execution stops, if the last state is one of the **accepting states**. If this is the case, we say the machine accepts the input word.

To learn how to program your own Turing Machines, read the tutorial.