A Turing machine cannot to tell if the program halt for each program-input pairs.

There is a subset of Turing machine that halt for experimental verification, so that it is possible a greater subset that can be verified in advance (that include the Turing machine experimentally verified).

If there is a finite length Turing machine program (the initial string on the tape of length L), and the Turing machine remain in the finite length L, then the Turing machine halt or loop, and it is possible to verify using two tapes, one with the output of the Turing machine, one other with the inner states of the Turing machine (for each output c_i a parallel cell with inner state s_i,or a cell with two outputs the binary value and the inner state), so that the possible configurations for a double cell are $2\mathcal{N}$ (where $\mathcal{N}$ are the possible inner state) for each step (two possible outputs, and $\mathcal{N}$ possible states) and the possible strings are $(2\mathcal{N})^L$: if it happen two identical strings of length L in two different step, then the Turing machine is in a loop, and the number of possible match are $(2\mathcal{N})^L$; the problem is solvable for Turing machine that write in a finite length L.

So that if the Turing machine write in a tape loop, then the Turing machine halt or loop, and it is possible verify the halting problem, and the maximum length of different symbols is $2\mathcal{N}$ (the tape loop length that can contain all the different outputs without cell repetitions).

A generic Turing machine can be verified only when the program is of finite length (the starting string with finite length L), if the Turing machine have a dynamic with two strings equal in different position on the tape then there is a loop (standing binary wave), if the program grows then the terminal string can have a periodicity: if there are only zeros on the tape beyond the program, then the head write in the terminal cells (inner state and output) two different symbols that can be compared with the previous symbols until the $(2\mathcal{N})!$ cells before (that is the maximum length of different symbols in a binary string) to verify a progressive binary wave on the left and on the right terminal part, if there are two waves, then the Turing machine does not halt.

So that only a infinite program cannot be verified.