

## What Can Turing Machines Do?

How powerful are Turing machines?

Turing machines can accept any regular or context-free language.

Turing machines can perform basic arithmetic computations.

Sequencing can be implemented on Turing machines.

Subroutine calls can be implemented on Turing machines.

---

### Turing's Thesis

Turing's Thesis states that any computation that can be carried out by “mechanical means” can be performed by a Turing machine (ignoring efficiency issues).

Turing's Thesis cannot be proved because a proof would require a precise definition of “mechanical means.”

Unfortunately, any definition of “mechanical means” would be another abstract model like Turing machines, and would not answer the question.

## Evidence for Turing's Thesis

1. Anything that can be done on any existing digital computer can also be done on a Turing machine.
  2. No one has yet been able to suggest a problem, solvable by what we intuitively consider an algorithm, that cannot be performed by a Turing machine.
  3. Alternative models have been proposed for mechanical computation, but none of them are more powerful than Turing machines.
- 

## Assuming Turing's Thesis

Turing's Thesis is not just an arbitrary assumption, but it is a scientific "law" akin to Peano's axioms, Newton's laws of motion, the periodic table, evolution, etc.

These laws or theories are not logically deduced, but they are plausible models that explain much of our world.

We accept them because the conclusions we draw from them agree with our experience and observations.

## Definition of Algorithm

An algorithm for a function  $f : D \rightarrow R$  is a Turing machine  $M$  such that if any  $d \in D$  is the input for  $M$ , then  $M$  eventually halts with  $f(d) \in R$  on the tape.

This is not a practical definition. Based on Turing's thesis, we can substitute any well-defined programming language for "Turing machine."