

Patna Women's College

MCA Department

Semester II

MCA CS2T07 Automata Theory Notes

Recursive language and Recursively enumerable languages

Recursive language

In [mathematics](#), [logic](#) and [computer science](#), a [formal language](#) (a [set](#) of finite sequences of [symbols](#) taken from a fixed [alphabet](#)) is called **recursive** if it is a [recursive subset](#) of the set of all possible finite sequences over the alphabet of the language. Equivalently, a formal language is recursive if there exists a [total Turing machine](#) (a [Turing machine](#) that halts for every given input) that, when given a finite sequence of symbols as input, accepts it if it belongs to the language and rejects it otherwise. Recursive languages are also called **decidable**.

The concept of **decidability** may be extended to other [models of computation](#). For example one may speak of languages decidable on a [non-deterministic Turing machine](#). Therefore, whenever an ambiguity is possible, the synonym for "recursive language" used is **Turing-decidable language**, rather than simply *decidable*.

There are two equivalent major definitions for the concept of a recursive language:

1. A recursive formal language is a [recursive subset](#) in the [set](#) of all possible words over the [alphabet](#) of the [language](#).
2. A recursive language is a formal language for which there exists a [Turing machine](#) that, when presented with any finite input [string](#), halts and accepts if the string is in the language, and halts and rejects otherwise. The Turing machine always halts: it is known as a [decider](#) and is said to *decide* the recursive language.

Recursively enumerable languages are known as **type-0** languages in the [Chomsky hierarchy](#) of formal languages. All [regular](#), [context-free](#), [context-sensitive](#) and [recursive](#) languages are recursively enumerable.

There are three equivalent definitions of a recursively enumerable language:

1. A recursively enumerable language is a [recursively enumerable subset](#) in the [set](#) of all possible words over the [alphabet](#) of the [language](#).
2. A recursively enumerable language is a formal language for which there exists a [Turing machine](#) (or other [computable function](#)) which will enumerate all valid strings of the language. Note that if the language is [infinite](#), the enumerating algorithm provided can be chosen so that it avoids repetitions, since we can test whether the string produced for number n is "already" produced for a number which is less than n . If it already is produced, use the output for input $n+1$ instead (recursively), but again, test whether it is "new".

3. A recursively enumerable language is a formal language for which there exists a Turing machine (or other computable function) that will halt and accept when presented with any **string** in the language as input but may either halt and reject or loop forever when presented with a string not in the language. Contrast this to **recursive languages**, which require that the Turing machine halts in all cases.

References:

https://en.wikipedia.org/wiki/Recursively_enumerable_language

https://en.wikipedia.org/wiki/Recursively_enumerable_language

Faculty:

Hera Shaheen

Assistant Professor

MCA Department

Patna Women's College