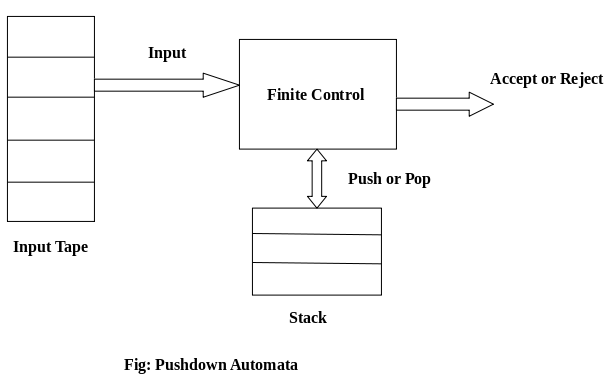
**Pushdown Automata (PDA)**

* Pushdown automata is a way to implement a CFG in the same way we design DFA for a regular grammar. A DFA can remember a finite amount of information, but a PDA can remember an infinite amount of information.
* Pushdown automata is simply an NFA augmented with an "external stack memory". The addition of stack is used to provide a last-in-first-out memory management capability to Pushdown automata. Pushdown automata can store an unbounded amount of information on the stack. It can access a limited amount of information on the stack.
* A PDA is more powerful than FA. Any language which can be acceptable by FA can also be acceptable by PDA. PDA also accepts a class of language which even cannot be accepted by FA. Thus, PDA is much more superior to FA.



## **PDA Components:**

**Input tape:** The input tape is divided in many cells or symbols. The input head is read-only and may only move from left to right, one symbol at a time.

**Finite control:** The finite control has some pointer which points the current symbol which is to be read.

**Stack:** The stack is a structure in which we can push and remove the items from one end only. It has an infinite size. In PDA, the stack is used to store the items temporarily

## **Formal definition of PDA:**

The PDA can be defined as a collection of 7 components:

**Q:** the finite set of states

**∑:** the input set

**Γ:** a stack symbol which can be pushed and popped from the stack

**q0:** the initial state

**Z:** a start symbol which is in Γ.

**F:** a set of final states

**δ:** mapping function which is used for moving from current state to next state.

# PDA Acceptance

A language can be accepted by Pushdown automata using two approaches:

**1. Acceptance by Final State:** The PDA is said to accept its input by the final state if it enters any final state in zero or more moves after reading the entire input.

**2. Acceptance by Empty Stack:** On reading the input string from the initial configuration for some PDA, the stack of PDA gets empty.

# Non-deterministic Pushdown Automata

The non-deterministic pushdown automata are very much similar to NFA. We will discuss some CFGs which accepts NPDA.

The CFG which accepts deterministic PDA accepts non-deterministic PDAs as well. Similarly, there are some CFGs which can be accepted only by NPDA and not by DPDA. Thus, NPDA is more powerful than DPDA.

# Turing Machine

Turing machine was invented in 1936 by **Alan Turing**. It is an accepting device which accepts Recursive Enumerable Language generated by type 0 grammar.

There are various features of the Turing machine:

1. It has an external memory which remembers arbitrary long sequence of input.
2. It has unlimited memory capability.
3. The model has a facility by which the input at left or right on the tape can be read easily.
4. The machine can produce a certain output based on its input. Sometimes it may be required that the same input has to be used to generate the output. So, in this machine, the distinction between input and output has been removed. Thus, a common set of alphabets can be used for the Turing machine.

## **Formal definition of Turing machine**

TM can be formally described as a 7-tuple (Q, ∑, Γ, δ, q0, B, F) where −

**Q** is a finite set of states

**∑** is the input alphabet (Not including B)

Γ is the tape alphabet(**∑+B)**

**q0** is the initial state

**B** is the blank symbol

**F** is the set of final states

**δ** is a transition function;

δ: Q × Γ → Q × X × {L, R}.

# Basic Model of Turing machine

A Turing Machine (TM) is a mathematical model which consists of a finite control, an input tape that is divided into cells and a read/write tape head that scans one cell at a time and can be moved to left or right.

Initially ‘n’ cells hold input of size ‘n’ .

The remaining infinity of cells hold a special symbol called Blank symbol (B)

After reading an input symbol, it is replaced with another symbol, its internal state is changed, and it moves from one cell to the right or left.

If the TM reaches the final state at the end of the input, the input string is accepted, otherwise rejected.

