

Top Down Parsing is a parsing technique in compiler design where the parser starts building the parse tree from the root (**start symbol**) and proceeds towards the leaves (**input symbols**).

It attempts to derive the given input string by repeatedly expanding the start symbol using production rules of the grammar.

There are two main types of top down parsers:

1. **Recursive Descent Parser** – Uses a set of recursive procedures to process the input.
2. **Predictive Parser (LL Parser)** – Uses lookahead to avoid backtracking.

**Advantages:**

- Simple and easy to implement.
- Works well for LL(1) grammars.
- Good for educational purposes and smaller compilers.