Programming Languages to form the arithmetic mathematical expressions. *The CFL is accepted by Push Down Automata (PDA)*

A **context-sensitive language** is generated by CSG. Union, intersection and concatenation of context-sensitive languages are also context-sensitive. *The CSL is accepted by Linear Bounded Automata (LBA).*

Please note that Complement of a context-sensitive language is also a CSL.

A language is recursive language (RL) if any or some ***Turing machine(TM) accepts it and halts on any input string.*** *The Recursive Language is accepted by Turing Machines (TM).*

A language is a recursively enumerable language (REL) if any or some ***Turing machine(TM) accepts it***. *The Recursive Enumerable Language is accepted by Turing Machines (TM).*

Closure Properties of Languages

Operation	REG	DCFL	CFL	CSL	RC	RE
Union	Yes	No	Yes	Yes	Yes	Yes
Intersection	Yes	No	No	Yes	Yes	Yes
Set difference	Yes	No	No	Yes	Yes	No
complementation	Yes	Yes	No	Yes	Yes	No
Intersection with a regular language	Yes	Yes	Yes	Yes	Yes	Yes
Union with a regular language	Yes	Yes	Yes	Yes	Yes	Yes
Left Difference with a regular languaqe (L-regular)	Yes	Yes	Yes	Yes	Yes	Yes
Right Difference with a regular language (Regular-L)	Yes	Yes	No	Yes	Yes	No
Concatenation	Yes	No	Yes	Yes	Yes	Yes
Kleene star	Yes	No	Yes	Yes	Yes	Yes
Kleene plus	Yes	No	Yes	Yes	Yes	Yes
Reversal	Yes	Yes	Yes	Yes	Yes	Yes
Substitution	Yes	No	Yes	No	No	Yes
Subset	No	No	No	No	No	No
Epsilon-free homomorphism	Yes	No	Yes	Yes	Yes	Yes

Homomorphism	Yes	No	Yes	No	No	Yes
Inverse homomorphism	Yes	Yes	Yes	Yes	Yes	Yes
Epsilon-free substitution	Yes	No	Yes	Yes	Yes	Yes
Right quotient with a reqular language	Yes	Yes	Yes	No	Yes	Yes
Left guotient with a reqular languaqe	Yes	Yes	Yes	No	Yes	Yes

The above table shows the closure property of languages against each operation. Either you can understand or you just by-heart this table that helps you to answer 1 or 2 questions.

> **❝** *Tip to remember easy: Either remember all Yes or remember all No. The table is divided in-to two portions 1. 'upto Subset is important', 2. others* **❞**

Phases of Compiler
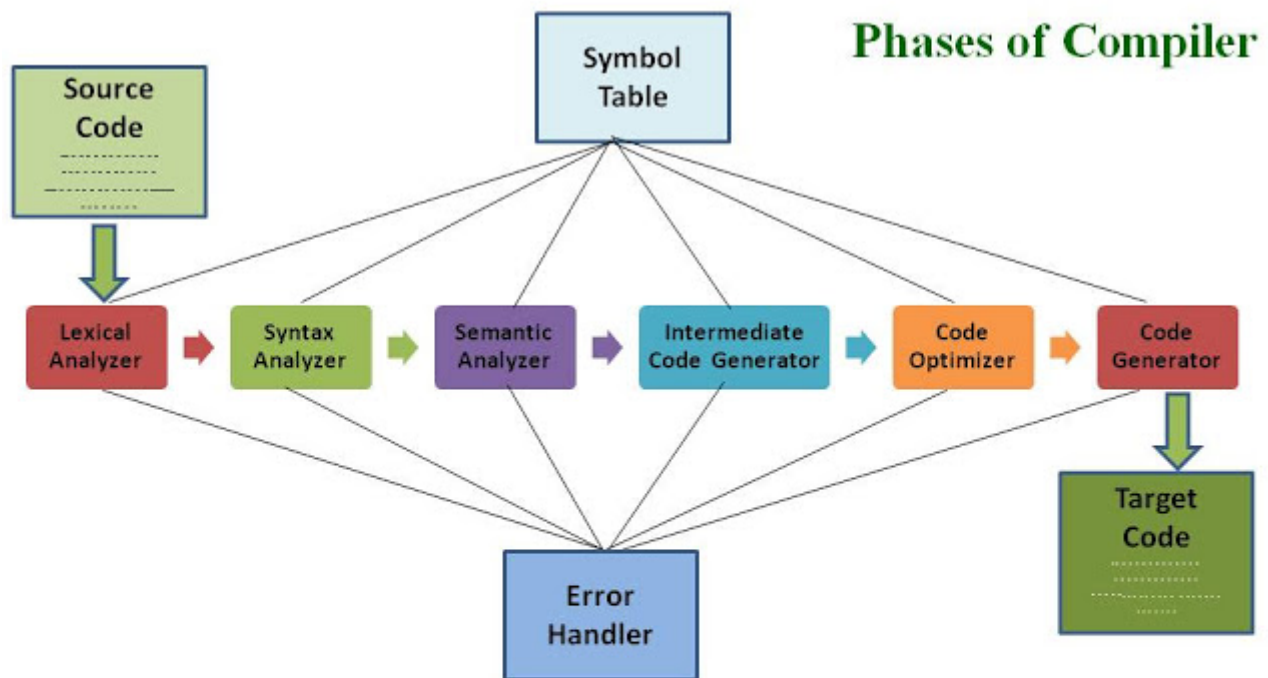
Phase	Lexical Analysis	Syntax Analysis	Semantic Analysis	Intermediate Code	Code Optimization	Code Generation
Output	Tokens	Parse Tree	Annotated Tree	Machine Independent code	Optimized Code	Target Code

Derivation: Derivation is the process of deriving the string from a given set of production rules. A parse tree (also called derivation tree) is a graphic re-presentation of a derivation that shows the hierarchical structure of the inputted language. Parse-tree is the output of the Syntax analysis phase.

Top-down Parsing: Top-down parser starts constructing the parse tree by initiating from the   ⌃ :