Semantic Analysis LiTeC

BATHINI ASHWITHA

KETHAVATH PRANEETH NAYAK

LANKA PRASANNA

MEGH SHAH

SANDEEP L

SIDDA BOENA JEEVAN SAMMESWAR

SAI SIVA ROHITH TIRUMALASETTI

What is semantic analysis?

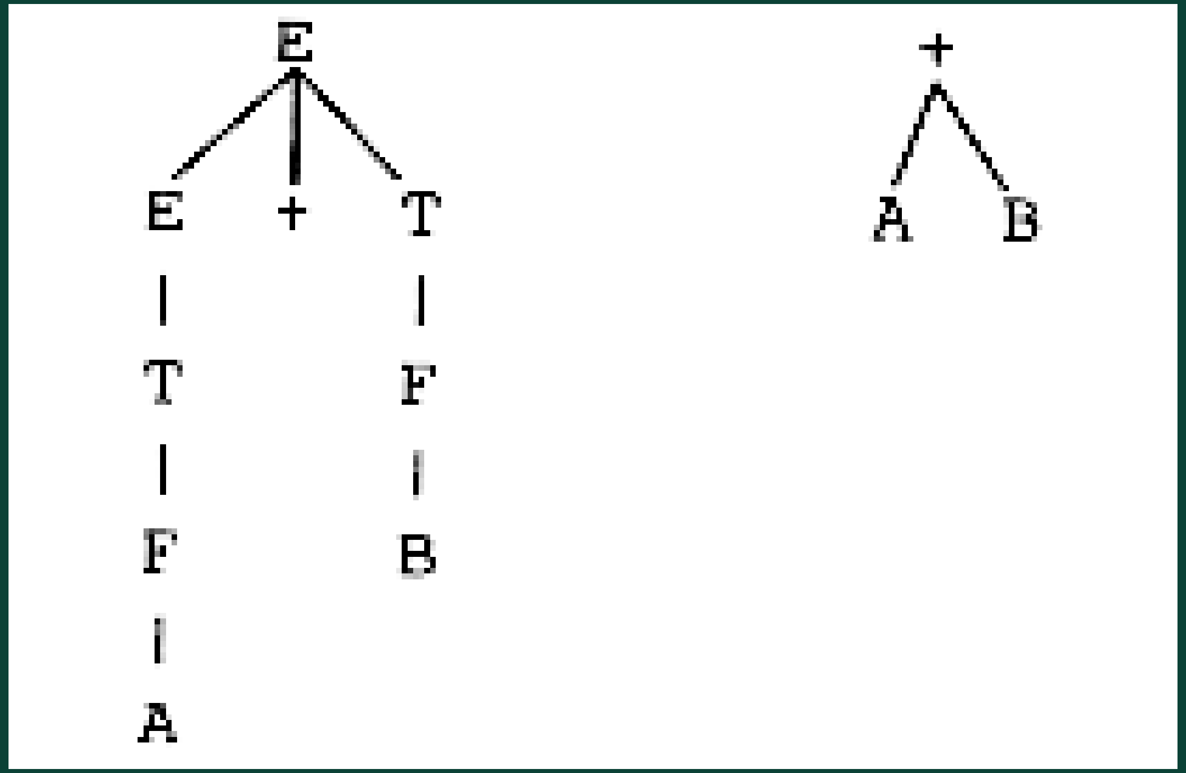
- Third phase of the compilation process.
- Checks whether the program is semantically consistent or not.
- Static semantics like type mismatch, re-declaration of variables, and undeclared variables.
- Syntax tree and symbol table are used to check the semantic rules of the given programming language.
- Semantic analyzer converts the parser's output to an annotated syntax tree.

Symbol table

- Important data structure maintained by compilers to store semantic information of variable names, function names, etc.
- Each symbol table entry contains the symbol name, attribute, type, and scope.
- Symbol table can be implemented using a linked list, binary search tree, or hash table.
- It is used for type checking in semantic analysis, increasing compile-time efficiency.
- Important operations of symbol table are insert( ) and lookup( ).

AST

- AST is a data structure that represents the entire program in a tree-like hierarchical structure.
- Abstract syntax tree is a simplified version of parse tree generated during syntax analysis without redundant information.
- Correctness of program can be verified by traversal of AST.



Attribute grammars

- Attribute grammars are an extension for context-free grammar.
- Semantic analyzers can be generated from attribute grammar.
- YACC permits only S- attributed grammar.
- It is used for passing values to nodes of the tree.
- It is used to validate semantic checks and can also be used to translate syntax trees to intermediate code.
- Static semantics of programming language can be specified using attribute grammar.

Semantic analysis of LiTeC

We started with the writing symbol table. We implemented a symbol table using hashing.

We created an abstract syntax tree. Added nodes to it from parser grammar

The following semantic checks were included in our code:

1. Re-declaration of variables.
2. Type checking for expressions and function return types.
3. Type checking of function arguments.
4. Undeclared variables.

THANK YOU