

REVIEW QUESTIONS

1. Define syntax and semantics.

- Syntax = form of expressions, statements, and program units
- Semantics = meaning of the expressions, statements, and program units

2. Who are language descriptions for?

- Initial evaluators
- Implementors
- Users

3. Describe the operation of a general language generator.

- Device that can be used to generate the sentences of the language
- Generates predictable sentences
- Limited usefulness as a language descriptor

4. Describe the operation of a general language recognizer.

- Device capable of reading strings of characters from alphabet
- Analyze given strings and either accept or reject based from language given
- Filters correct and incorrect sentences

5. What is the difference between a sentence and a sentential form?

- Sentence = a sentential form that only has terminal symbols
- Sentential Form = every string of symbols in the derivation

6. Define a left-recursive grammar rule.

- When a grammar rule has its LHS also appearing at the beginning of its RHS

7. What three extensions are common to most EBNFs?

-

8. Distinguish between static and dynamic semantics.

- Static Semantics = indirectly related to the meaning of programs during execution. Focuses more on the legal forms of programs (syntax). Checked at compile time
- Dynamic Semantics = describing the meaning of the programs. Precise knowledge what statements do. Compile writers determine the semantics of a language

9. What purpose do predicates serve in an attribute grammar?

- State the static semantic rules of the language

10. What is the difference between a synthesized and an inherited attribute?

- Synthesized Attribute = result of the attribute evaluation rules. May use the inherited attribute values. Pass semantic information up the parse tree

- Inherited Attribute = Passed down from parent nodes. Pass semantic information down the parse tree

11. How is the order of evaluation of attributes determined for the trees of a given attribute grammar?

- Based on underlying BNF grammar
- Possible empty set of attribute values attached to each node

12. What is the primary use of attribute grammars?

- Describe more of the structure of a programming language than can be described with a context-free grammar

13. Explain the primary uses of a methodology and notation for describing the semantics of programming languages.

- Programmers need to know what statements do to use them effectively
- Compiler writers must know what language constructs mean to correctly implement designs
- Language designers should develop the semantic descriptions of the language and discover ambiguities and inconsistencies in their design in the process

14. Why can machine languages not be used to define statements in operational semantics?

- Steps in machine language execution and changes to the state and machine are small and numerous.
- Storage of computing machines is too large and complex. Several levels of memory devices

15. Describe the two levels of uses of operational semantics.

- Highest Level = interest in the final result of the execution of a complete program. "Natural Operational Semantics"
- Lowest Level = determine the precise meaning of a program through an examination of the complete sequence of state changes during execution. "Structural Operational Semantics"

16. In denotational semantics, what are the syntactic and semantic domains?

- Domain (Syntactic Domain) = collection of values that are genuine parameters to a function
- Range (Semantic Domain) = collection of objects to which the parameters are mapped

17. What is stored in the state of a program for denotational semantics?

- Value of all current variables

18. Which semantics approach is most widely known?

- Denotational Semantics

19. What two things must be defined for each language entity in order to construct a denotational description of the language?

- Objects
- Functions

20. Which part of an inference rule is the antecedent?

- Antecedent = Top part

21. What is a predicate transformer function?

- $Wp(\text{statement}, \text{post condition}) = \text{precondition}$
- A wp function is called a predicate transformer. Takes a predicate as a parameter and returns another predicate

22. What does partial correctness mean for a loop construct?

- When termination is not guaranteed

23. On what branch of mathematics is axiomatic semantics based?

- Mathematics Logic

24. On what branch of mathematics is denotational semantics based?

- Mathematical Objects (Denotations)

25. What is the problem with using a software pure interpreter for operational semantics?

- Detailed characteristics of the particular computer would make actions difficult to understand.
- Machine-dependent semantic definition

26. Explain what the preconditions and postconditions of a given statement mean in axiomatic semantics.

- Assertion (Precondition) = assertion before a statement states the relationships and constraints among variables that are true at a certain point in execution.
- Postcondition = assertion following a statement

27. Describe the approach of using axiomatic semantics to prove the correctness of a given program.

- Specifies what can be proven about the program
- States precisely the meaning of the statements and programs in terms of logic expressions

28. Describe the basic concept of denotational semantics.

- Mapping every syntactic entity associated with a programming language into some form of mathematical entity, translating programming language constructs into mathematical objects

29. In what fundamental way do operational semantics and denotational semantics differ?

- Operational Semantics = state changes are defined by coded algorithms for a virtual machine
- Denotational Semantics = rigorous mathematical functions