

24. What is an example of two language design criteria that are in direct conflict with each other?
25. What are the three general methods of implementing a programming language?
26. Which produces faster program execution, a compiler or a pure interpreter?
27. What role does the symbol table play in a compiler?
28. What does a linker do?
29. Why is the von Neumann bottleneck important?
30. What are the advantages in implementing a language with a pure interpreter?

PROBLEM SET

1. Do you believe our capacity for abstract thought is influenced by our language skills? Support your opinion.
2. What are some features of specific programming languages you know whose rationales are a mystery to you?
3. What arguments can you make for the idea of a single language for all programming domains?
4. What arguments can you make against the idea of a single language for all programming domains?
5. Name and explain another criterion by which languages can be judged (in addition to those discussed in this chapter).
6. What common programming language statement, in your opinion, is most detrimental to readability?
7. Java uses a right brace to mark the end of all compound statements. What are the arguments for and against this design?
8. Many languages distinguish between uppercase and lowercase letters in user-defined names. What are the pros and cons of this design decision?
9. Explain the different aspects of the cost of a programming language.
10. What are the arguments for writing efficient programs even though hardware is relatively inexpensive?
11. Describe some design trade-offs between efficiency and safety in some language you know.
12. What major features would a perfect programming language include, in your opinion?
13. Was the first high-level programming language you learned implemented with a pure interpreter, a hybrid implementation system, or a compiler? (You would not necessarily know this without research.)

14. Describe the advantages and disadvantages of some programming environment you have used.
15. How do type declaration statements for simple variables affect the readability of a language, considering that some languages do not require them?
16. Write an evaluation of some programming language you know, using the criteria described in this chapter.
17. Some programming languages—for example, Pascal—have used the semicolon to separate statements, while Java uses it to terminate statements. Which of these, in your opinion, is most natural and least likely to result in syntax errors? Support your answer.
18. Many contemporary languages allow two kinds of comments: one in which delimiters are used on both ends (multiple-line comments), and one in which a delimiter marks only the beginning of the comment (one-line comments). Discuss the advantages and disadvantages of each of these with respect to our criteria.

50. For what application area is JavaScript most widely used?
51. What is the relationship between JavaScript and PHP, in terms of their use?
52. PHP's primary data structure is a combination of what two data structures from other languages?
53. What data structure does Python use in place of arrays?
54. What characteristic does Ruby share with Smalltalk?
55. What characteristic of Ruby's arithmetic operators makes them unique among those of other languages?
56. What data structures are built into Lua?
57. Is Lua normally compiled, purely interpreted, or impurely interpreted?
58. What feature of Delphi's classes is included in C#?
59. What deficiency of the **switch** statement of C is addressed with the changes made by C# to that statement?
60. What is the primary platform on which C# is used?
61. What are the inputs to an XSLT processor?
62. What is the output of an XSLT processor?
63. What element of the JSTL is related to a subprogram?
64. To what is a JSP document converted by a JSP processor?
65. Where are servlets executed?

PROBLEM SET

1. What features of Plankalkül do you think would have had the greatest influence on Fortran 0 if the Fortran designers had been familiar with Plankalkül?
2. Determine the capabilities of Backus's 701 Speedcoding system, and compare them with those of a contemporary programmable hand calculator.
3. Write a short history of the A-0, A-1, and A-2 systems designed by Grace Hopper and her associates.
4. As a research project, compare the facilities of Fortran 0 with those of the Laning and Zierler system.
5. Which of the three original goals of the ALGOL design committee, in your opinion, was most difficult to achieve at that time?
6. Make an educated guess as to the most common syntax error in LISP programs.
7. LISP began as a pure functional language but gradually acquired more and more imperative features. Why?

8. Describe in detail the three most important reasons, in your opinion, why ALGOL 60 did not become a very widely used language.
9. Why, in your opinion, did COBOL allow long identifiers when Fortran and ALGOL did not?
10. Outline the major motivation of IBM in developing PL/I.
11. Was IBM's assumption, on which it based its decision to develop PL/I correct, given the history of computers and language developments since 1964?
12. Describe, in your own words, the concept of orthogonality in programming language design.
13. What is the primary reason why PL/I became more widely used than ALGOL 68?
14. What are the arguments both for and against the idea of a typeless language?
15. Are there any logic programming languages other than Prolog?
16. What is your opinion of the argument that languages that are too complex are too dangerous to use, and we should therefore keep all languages small and simple?
17. Do you think language design by committee is a good idea? Support your opinion.
18. Languages continually evolve. What sort of restrictions do you think are appropriate for changes in programming languages? Compare your answers with the evolution of Fortran.
19. Build a table identifying all of the major language developments, together with when they occurred, in what language they first appeared, and the identities of the developers.
20. There have been some public interchanges between Microsoft and Sun concerning the design of Microsoft's J++ and C# and Sun's Java. Read some of these documents, which are available on their respective Web sites, and write an analysis of the disagreements concerning the delegates.
21. The scripting languages have in recent years evolved data structures to replace traditional arrays. Explain the chronological sequence of these developments.
22. Explain two reasons why pure interpretation is an acceptable implementation method for several recent scripting languages.
23. Perl 6, when it arrives, will likely be a significantly enlarged language. Make an educated guess as to whether a language like Lua will also grow continuously over its lifetime. Support your answer.
24. Why, in your opinion, do new scripting languages appear more frequently than new compiled languages?
25. Give a brief general description of a markup/programming hybrid language.