



TEXTBOOK TESTBANKS

Test Bank for Programming Languages: Concepts and Implementation 1st Perugini

TEST BANK SAMPLE PREVIEW

PUBLISHER

JBLearning

COPYRIGHT

2023

RESOURCE TYPE

Test Bank

ISBN

9781284222722

*Test Bank for Programming
Languages-Concepts and
Implementation 1st Edition by Perugini*

ISBN: 9781284222722

Import Settings:
Base Settings: Brownstone Default
Information Field: Complexity
Information Field: Ahead
Information Field: Subject
Information Field: Title
Highest Answer Letter: E
Multiple Keywords in Same Paragraph: No

Chapter: Chapter 1 - Quiz

Multiple Choice

1. Which of the following is *not* an aspect of the *imperative* style of programming?

- A) First-class functions
- B) Assignment statements
- C) Sequential execution
- D) Variables

Ans: A

Complexity: Moderate

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

2. Which of the following is *not* a criterion on which to evaluate programming languages?

- A) Readability
- B) Writability
- C) Linkability
- D) Portability
- E) Usability

Ans: C

Complexity: Easy

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

3. Scheme primarily supports _____ programming, but has some support for _____ programming.

- A) imperative; functional
- B) functional; imperative

Ans: B

Complexity: Moderate

Ahead: 1.4 Styles of Programming

Subject: 1

Title: Introduction

True/False

1. True or False? The value of a variable is bound statically.

Ans: False

Complexity: Easy

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

2. True or False? The memory address of a local variable is bound statically.

Ans: False

Complexity: Moderate

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

3. True or False? The memory address of a global variable is bound statically.

Ans: True

Complexity: Moderate

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

4. True or False? Statements and expressions are synonymous in programming languages (i.e., they refer to the same thing).

Ans: False

Complexity: Easy

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

5. True or False? Functions are *first-class* in languages supporting functional programming.

Ans: True

Complexity: Easy

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

6. True or False? Objects are *first-class* in C++.

Ans: True

Complexity: Moderate

Ahead: 1.4 Styles of Programming
Subject: Chapter 1
Title: Introduction

7. True or False? Objects are *first-class* in Java.

Ans: True

Complexity: Moderate

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

8. True or False? The concept of a language *paradigm* is antiquated.

Ans: True

Complexity: Easy

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

9. True or False? It is possible for a language with support for *pure* functional programming to have an assignment operator.

Ans: False

Complexity: Moderate

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

10. True or False? Any function that is without *side effect* is *referentially transparent*.

Ans: False

Complexity: Difficult

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

11. True or False? Any function that is *referentially transparent* is without *side effect*.

Ans: False

Complexity: Difficult

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

12. True or False? Speed of program development is no longer as important a factor in the design of programming languages as it once was.

Ans: False

Complexity: Easy

Ahead: 1.5 Factors Influencing Language Development
Subject: Chapter 1
Title: Introduction

Short Answer

1. Which concept of programming languages is fundamental in the sense that other concepts can be viewed as instances of this concept?

Ans: binding

Complexity: Moderate

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

2. The *imperative* style of programming is based on which computer architecture?

Ans: von Neumann architecture

Complexity: Easy

Ahead: 1.4 Styles of Programming

Subject: Chapter 1

Title: Introduction

Essay

1. What is a *programming language*?

Ans: A programming language is a system of data-manipulation rules for describing computation.

Complexity: Easy

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

2. List five binding times in the study of programming languages, arrange them in order from earliest to latest.

Ans: language definition time, language implementation time, compile time, link time, load time, and run-time

Complexity: Moderate

Ahead: 1.3 The World of Programming Languages

Subject: Chapter 1

Title: Introduction

3. List two other styles of programming other than the familiar imperative and object-oriented styles of programming.

Ans: functional programming, logic; declarative programming

Complexity: Easy
Ahead: 1.4 Styles of Programming
Subject: Chapter 1
Title: Introduction