Your Results for: " Capitulo 14"

Print this page

Book Title: C++ como Programar
Book Author: Deitel
Location on Site: Exercícios de múltipla escolha (em Inglês) > Capitulo 14
Date/Time Submitted: December 5, 2013 at 10:04 AM (UTC/GMT)

Summary of Results

100% Correct of 13 Scored items:
13 Correct: 100%
0 Incorrect: 0%

More information about scoring

---

1. CORRECT

Section 14.1 Introduction

14.1 Q1: The relationship between function templates and function-template specializations is most similar to the relationship between:

Your Answer: Classes and objects.

---

2. CORRECT

Section 14.2 Function Templates

14.2 Q1: A difference between function-template specializations and overloaded functions is that:

Your Answer: Function-template specializations are generated by the compiler, not the programmer.

---

3. CORRECT

14.2 Q2: Function-template specializations:

Your Answer: Are generated at compile time.

---

4. CORRECT

Section 14.3 Overloading Function Templates

14.3 Q1: A function template can be overloaded by:

Your Answer: Using non-template functions with the same name and different parameters.

---

5. CORRECT

14.3 Q2: Assuming that all four of the following functions are defined, which one will be called by the function call square(23.4)?

Your Answer: double square(double num).

---

6. CORRECT

Section 14.4 Class Templates

14.4 Q1: Class templates:

Your Answer: Must put template< typename Type > before the class definition.

---

7. CORRECT 14.4 Q2: For a class template, the binary scope resolution operator (::) is needed:

Your Answer: Only in the definitions of the member functions defined outside the class.

---

8. CORRECT 14.4 Q3: Function templates:

Your Answer: Can include objects of template classes as parameters.

---

9. CORRECT Section 14.5 Nontype Parameters and Default Types for Class Templates

14.5 Q1: Nontype parameters are:

Your Answer: const.

---

10. CORRECT 14.5 Q2: Default type parameters are allowed only:

Your Answer: As the rightmost (trailing) parameters in a template's type-parameter list.

---

11. CORRECT Section 14.6 Notes on Templates and Inheritance

14.6 Q1: Select the incorrect statement.

Your Answer: A non-template class can be used to derive a class-template specialization.

---

12. CORRECT Section 14.7 Notes on Templates and Friends

14.7 Q1: Friendship cannot be declared between a class template and:

Your Answer: Another class template.

---

13. CORRECT Section 14.8 Notes on Templates and static Members

14.8 Q1: Which of the following is false?

Your Answer: One copy of each static member function is shared between all class-template specializations in the class template.

E-mail Your Results

My name is (first last):

E-mail my results to:

	E-mail address:	Send as:
☐ Me		Text
☐ Instructor		Text
☐ TA		Text
☐ Other		Text

Help E-mail Results