Home ➤ Exercícios de múltipla escolha (em Inglês) ➤ Capitulo 24

Capitulo 24

This activity contains 19 questions.

1.

Section 24.2 const_cast Operator

24.2 Q1: The const_cast operator can be used to cast away ____________ or _____________ qualifications.

○ *Property, const.*

○ *enum, #define.*

○ *const, volatile.*

○ *The const-ness property, remove.*

2.

24.2 Q2: The const_cast operator is needed when ___________ data must be treated as ____________ data.

○ *const, const.*

○ *Non-const, const.*

○ *Non-const, non-const.*

○ *const, non-const.*

3.

Section 24.3 namespaces

24.3 Q1: ____________ is/are used to solve the problem of variables with the same name and overlapping scopes.

○ *Dynamic memory allocation.*

○ *Classes.*

○ *Casts.*

○ *namespaces.*

4.

24.3 Q2: Without the statement ___________, all ___________ statements must be written as std::cout.

○ *using std;, namespace std::cout.*

○ *using std::namespace;, cout.*

○ *using namespace std;, cout.*

○ *using std namespace;, cout.*

5.

24.3 Q3: namespace definitions are different from class definitions because:

- ○ *namespace definitions do not end in semicolons.*
- ○ *namespaces cannot contain functions.*
- ○ *namespaces cannot contain variables.*
- ○ *namespace definitions are not delimited by braces {}.*

6.

24.3 Q4: _____________ namespace members appear to occupy the ____________ namespace and do not have to be qualified with a namespace name.

- ○ *Named, global.*
- ○ *Global, unnamed.*
- ○ *Named, static.*
- ○ *Unnamed, global.*

7.

24.3 Q5: Individual namespace members can be accessed without a namespace qualifier:

- ○ *If the using namespace all; directive is included at the beginning of the file.*
- ○ *At any time.*
- ○ *Only if the namespace is declared inside of main.*
- ○ *If the individual member is mentioned in a using namespace::namespacemember declaration.*

8.

24.3 Q6: namespaces cannot contain:

- ○ *main.*
- ○ *Other namespaces.*
- ○ *Classes.*
- ○ *Any functions.*

9.

Section 24.4 Operator Keywords

24.4 Q1: Which operator keyword corresponds to ^?

- ○ *xor_eq.*

○ *xor.*

○ *or.*

○ *or_eq.*

10. *24.4 Q2: Which operator corresponds to operator keyword and_eq?*

○ *&&.*

○ *!=.*

○ *&.*

○ *&=.*

11. *Section 24.5 mutable Class Members*

24.5 Q1: If a certain object's data members need to be frequently modified by const functions, it is best to:

○ *Declare the member values non-const.*

○ *Use const_cast to make the member values modifiable.*

○ *Declare the member values mutable.*

○ *Use static_cast to make the member values modifiable.*

12. *Section 24.6 Pointers to Class Members (.* and ->*)*

24.6 Q1: Which symbol is used to access class members via pointers to class members?

○ *..*

○ **..*

○ *.*.*

○ *->.*

13. *24.6 Q2: Which of the following declares a pointer ptr to a class member function in class Check that takes an int argument and returns a bool?*

○ *bool*(int) Check::ptr;.*

○ *bool (Check::*ptr)(int);.*

○ *Check::(bool (*ptr)(int));.*

○ *bool:(int) (*Check::ptr);.*

14.

24.6 Q3: Assuming that ptr is a pointer to a class data member of class Check and obj is a pointer to a Check object, which of the following statements would be used to output that data member?

○ cout << obj->*ptr();.

○ cout << (*obj).*ptr;.

○ cout << obj.*ptr;.

○ cout << (*obj)->*ptr;.

15.

Section 24.7 Multiple Inheritance

24.7 Q1: Multiple inheritance means that a derived class inherits the members of several:

○ public classes.

○ Derived classes.

○ Indirect base classes, but exactly one direct base class.

○ Direct base classes.

16.

24.7 Q2: If a derived class inherits from two base classes that both define a certain member function, then an ambiguity problem will arise when:

○ An object of the derived class calls that certain member function.

○ None of the above.

○ An object of a base class calls that certain member function.

○ A pointer of a base class type pointing to an object of the derived class calls that certain member function.

17.

Section 24.8 Multiple Inheritance and virtual Base Classes

24.8 Q1: Duplicate __________ could result from ___________.

○ subclasses, upcasting a pointer.

○ superclasses, downcasting a pointer.

○ subobjects, multiple inheritance.

○ superobjects, multiple inheritance.

18. *24.8 Q2: ___________ inheritance solves the problem of duplicate subobjects.*

○ *virtual.*

○ *const.*

○ *static.*

○ *protected.*

19. *24.8 Q3: For multiple inheritance, the _______________ is responsible for initializing the __________ base class.*

○ *Least derived class, virtual.*

○ *Most derived class, virtual.*

○ *First derived class, virtual.*

○ *Most derived class, const.*

Clear Answers / Start Over   Submit Answers for Grading

Answer choices in this exercise appear in a different order each time the page is loaded.