

Operators and Overloading

Solutions

- Give an example of an operator for a built-in type
+ operator to add two ints
- How is your operator called?
a + b; // calls operator +(a, b)

- Give an example of an operator for a type from the C++ Standard Library

+ operator to add two std::strings

- How is your operator called?

a + b; // calls operator +(a, b)

- What is a unary operator?

A unary operator takes one argument

- What is a binary operator?

A binary operator takes two arguments

- Give an example of an operator that can be either unary or binary

The minus operator can take one argument (sign inversion) or two (subtraction)

Which of the following statements are true?

- A) Operators are "overloaded" when they have the same name and take the same arguments
- B) All operators can take either one or two arguments
- C) Operators can be member functions or non-member functions
- D) Operators which act on classes do not need to have the same syntax as operators which act on built-in types

C is true

Which of the following statements are true?

- A) When we write our own classes, we can add C++ operators to them
- B) Operators that we define ourselves can only use the same symbols as the built-in types
- C) However, we need not follow the same syntax rules
- D) When writing an operator, we should be creative and not restrict ourselves to the semantics of built-in operators

A and B are true