Advanced Programming

Operator Overloading

Topics

- Operators in C++
- Overloading Operators
  - Which operators can be loaded?
  - When can we overload an operator?
  - How can we overload an operator?
- Examples

Operators

- Operators define:
  - The operation to be carried out
  - The number of operands used in the operation
  - The type of the operands
  - The type of the resulted value
  - The precedence of the operation in an expression

Properties of Operators

- If the type of the operands are not the expected types, they are converted (cast) into the required format
- A symbol may be used for different operations, or operations with different types of operands, or different number of operands

Overloading Operators

- The operation of an operand can be defined for user defined variables (classes)
- However:
  - New symbols (operators) cannot be defined
  - Number of operands for an operator cannot be changed
  - The precedence of an operator cannot be changed

Overloading Operators

- Operators are overloaded using a function which can be
  - A member function in a class
  - A general function

Example 1

- Define a class for complex numbers and overload + and – operators to add and subtract two complex numbers

Example 2

- Overload * to multiply two matrices (Define a Matrix class)
- Overload * operator to left multiply a matrix by a scalar
- Overload * operator to right multiply a matrix by a scalar

Example 3

- Define a class named vector2D to represent 2D vectord
- Overload + and – operators to add and subtract vectors
- Overload ++ operator to increment the value of a vector2D by incrementing its x and y components by one

Example 4

- Define a class for integer arrays which can check the boundaries of the array indices.
- Overload [] operators to check the range of the indices.