Prefix and Postfix Operators Solutions

Prefix and Postfix Operators

- Describe the functioning of the prefix ++ operator and the postfix ++ operator
  - The prefix ++ operator increments its argument and returns a reference to the incremented object
  - The postfix ++ operator increments its argument and returns a copy of the original object

++ Operators and Arrays

- If p is a pointer to an element in an array, what happens when p is incremented?
  - p will now be a pointer to the next element in the array

Postfix ++ Operator

- Imagine p is a pointer to the first element in an array
- Explain what this code does

```
cout << *p++ << endl;
```

  - p is incremented, because ++ has a higher operator precedence than *
  - p now points to the second element in the array
  - The postfix operator returns a pointer to the first element
  - The * operator dereferences this pointer
  - The code will print out the first element in the array

Postfix ++ Operator and Loops

- What is the difference between these two ways of writing a loop?

```
for (int i = 0; i < 4; ++i) { ... }
for (int i = 0; i < 4; i++) { ... }
```

  - The postfix makes a copy of the loop counter, which is less efficient
  - The prefix version should be preferred

Prefix and Postfix Operators

- What are the prototypes of the prefix and postfix ++ operators?
  - Test& Test::operator ++(); // Prefix
  - Test Test::operator ++(int t); // Postfix

Basic Complex Number Class

- Write a definition of a simple class that implements both forms of the ++ and -- operators
- Write a program to test your operators and check that they give the correct answers