

# Prefix and Postfix ++ Operators

## Prefix and Postfix ++ Operators

- The prefix operator is called when the ++ is written before the variable being incremented

```
++p;
```

- The postfix operator is called when the ++ is written after the variable being incremented

```
p++;
```

- There are also decrement versions:

```
--p;           // Prefix -- operator
```

```
p--;           // Postfix -- operator
```

## Overloading Prefix ++Operator

- If we are writing a class that supports arithmetical operations, we need to implement these operators
- The prefix operator performs the increment and returns the incremented value

```
type& type::operator ++() {  
    member = member + 1;           // Perform the increment operation  
    return *this;                  // Return the incremented instance value  
}
```

- The prefix -- operator is implemented similarly

## Overloading Postfix ++ Operator

- The postfix operator performs the increment and returns the original value
- To do this, it must make a copy of the original value
- The postfix operator takes a dummy argument, to give it a different signature from the prefix operator

```
type type::operator ++(int t) {           // Dummy argument
    type temp(*this);                     // Copy the current instance value
    member = member + 1;                  // Perform the increment operation
    return temp;                          // Return the original instance value
}
```