

UNARY POSITIVE AND NEGATIVE OPERATORS*

Kenneth Leroy Busbee

This work is produced by OpenStax-CNX and licensed under the Creative Commons Attribution License 3.0[†]

Abstract

An introduction to unary positive and unary negative operators used in programming with details of C++ usage.

1 General Discussion

Unary positive also known as plus and unary negative also known as minus are unique operators. The plus and minus when used with a constant value represent the concept that the values are either positive or negative. Let's consider:

+5 + -2

We have three operators in this order: unary positive, addition, and unary negative. The answer to this expression is a positive 3. As you can see, one must differentiate between when the plus sign means unary positive and when it means addition. Unary negative and subtraction have the same problem. Let's consider:

-2 - +5

The expression evaluates to negative 7. Let's consider:

7 - -2

First constants that do not have a unary minus in front of them are assumed (the default) to be positive. When you subtract a negative number it is like adding, thus the expression evaluates to positive 9.

2 C++ Code Examples

The above examples work within the C++ programming language. What happens if we put a unary positive or unary negative in front of a variable or a named constant?

2.1 Negation – Unary Negative

The concept of negation is to take a value and change its sign, that is: flip it. If it positive make it negative and if it is negative make it positive. Mathematically, it is the following C++ code example, given that money is an integer variable with a value of 6:

```
-money
money * -1
```

*Version 1.2: Jan 15, 2010 8:52 am -0600

[†]<http://creativecommons.org/licenses/by/3.0/>

The above two expressions evaluate to the same value. In the first line, the value in the variable `money` is fetched and then it's negated to a negative 6. In the second line, the value in the variable `money` is fetched and then it's multiplied by negative 1 making the answer a negative 6.

2.2 Unary Positive – Worthless

Simply to satisfy symmetry, the unary positive was added to the C++ programming language as an operator. However, it is a totally worthless or useless operator and is rarely used. However don't be confused the following expression is completely valid:

```
6 + +5
```

The second `+` sign is interpreted as unary positive. The first `+` sign is interpreted as addition.

```
money
```

```
+money
```

```
money * +1
```

For all three lines, if the value stored in `money` is 6 the value of the expression is 6. Even if the value in `money` was negative 77 the value of the expression would be negative 77. The operator does nothing, because multiplying anything by 1 does not change its value.

2.3 Possible Confusion

Do not confuse the unary negative operator with decrement. Decrement changes the value in the variable and thus is an Lvalue concept. Unary negative does not change the value of the variable, but uses it in an Rvalue context. It fetches the value and then negates that value. The original value in the variable does not change.

Because there is no changing of the value associated with the identifier name, the identifier name could represent a variable or named constant.

2.4 Exercises

Exercise 1

(Solution on p. 3.)

Evaluate the following items involving unary positive and unary negative:

1. `+10 - -2`
2. `-18 + 24`
3. `4 - +3`
4. `+8 + - +5`
5. `+8 + / +5`

3 Definitions

Definition 1: unary positive

A worthless operator almost never used.

Definition 2: unary negative

An operator that causes negation.

Definition 3: plus

Aka unary positive.

Definition 4: minus

Aka unary negative.

Solutions to Exercises in this Module

Solution to Exercise (p. 2)

Answers:

1. 12
2. 6
3. 1
4. It's 3. Surprised, but it works. The middle plus sign is addition and the rest are unary positive or unary negative.
5. Error, no operand between addition and division.