Document number: N3472
Date: 2012-10-19
Project: Programming Language C++
Reply to: James Dennett
jdennett@google.com

Binary Literals in the C++ Core Language

Overview

EWG decided in Portland (October 2012) to add core language support for binary literals, with a 0b/0B prefix to distinguish them (just as we have 0x/0X for hex and 0 for octal today). This was after guidance from LWG that they preferred not to add this support to the library.

Use of an 0b/0B prefix for binary literals is an existing GCC extension (also supported by Clang), and is the same syntax as Java 7, Python, and D.

As with all other integer literal forms, binary integer literals will support user-defined suffixes.

It works well with the digit separator proposal, allowing writing 0b1000_1111. The proposed wording here assumes that allowing underscore as a digit separator is covered elsewhere; it's likely that the edits here might conflict (hopefully, in benign ways) with edits there.

There is not a proposal to allow a new *binary-escape-sequence*. \b is taken in any case.

References

Descriptions of
http://gcc.gnu.org/onlinedocs/gcc/Binary-constants.html
http://www.digitalmars.com/d/1.0/lex.html#Integer
http://docs.python.org/release/3.2.3/reference/lexical_analysis.html#integer-literals
http://docs.oracle.com/javase/7/docs/technotes/guides/language/binary-literals.html

Proposed Wording

(New text is red and underlined, deleted text ~~is in strikethrough~~.)

2.14.2 Integer literals

integer-literal:
decimal-literal integer-suffix$_{opt}$

octal-literal integer-suffix$_{opt}$
hexadecimal-literal integer-suffix$_{opt}$
binary-literal integer-suffix$_{opt}$

decimal-literal:
nonzero-digit
decimal-literal digit

octal-literal:
0
octal-literal octal-digit

hexadecimal-literal:
0x *hexadecimal-digit*
0X *hexadecimal-digit*
hexadecimal-literal hexadecimal-digit

binary-literal:
0b *binary-digit*
0B *binary-digit*
binary-literal binary-digit

nonzero-digit: one of
1 2 3 4 5 6 7 8 9

octal-digit: one of
0 1 2 3 4 5 6 7

hexadecimal-digit: one of
0 1 2 3 4 5 6 7 8 9
a b c d e f
A B C D E F

binary-digit:
0
1

2.14.2 Integer Literals [lex.icon]

1. An *integer literal* is a sequence of digits that has no period or exponent part. An integer literal may have a prefix that specifies its base and a suffix that specifies its type. The lexically first digit of the sequence of digits is the most significant. A *decimal* integer literal

(base ten) begins with a digit other than 0 and consists of a sequence of decimal digits. An *octal* integer literal (base eight) begins with the digit 0 and consists of a sequence of octal digits. A *hexadecimal* integer literal (base sixteen) begins with 0x or 0X and consists of a sequence of hexadecimal digits, which include the decimal digits and the letters a through f and A through F with decimal values ten through fifteen. A *binary* integer literal (base two) begins with 0b or 0B and consists of a sequence of binary digits. [*Example:* the number twelve can be written 12, 014, ~~or~~ 0XC, or 0b1100. — *end example*]

2.14.8 User-defined literals [lex.ext]

user-defined-integer-literal:
decimal-literal ud-suffix
octal-literal ud-suffix
hexadecimal-literal ud-suffix
binary-literal ud-suffix

A.2 Lexical conventions [gram.lex]

Should be modified to reflect the above changes as usual.