

## C/C++ Tip 0001 — Escape Sequences in Header Names

© 2009 Rex Jaeschke. All rights reserved.

**Issue:** I program in an environment using a hierarchical file system in which path names have the form “\level1\level2\filename.ext”. Given that the path for a user-defined header is \root\myheader.h, what are the rules governing the following preprocessor directive?

```
#include "\root\myheader.h"
```

Specifically, are \r and \m treated as a well-defined and undefined escape sequence, respectively?

**Response:** The underlying question here is “Are escape sequences recognized in #include directives having this form?” The simple answer is “Maybe”.

According to C99, §6.4.7, “Header names”, the relevant syntax for a *header-name* token is

*header-name:*

" *q-char-sequence* "

*q-char-sequence:*

*q-char*

*q-char-sequence q-char*

*q-char:*

any member of the source character set except the new-line character and "

As such, the stream of characters inside the double quotes is *not* recognized as a string literal.

The C Standard goes on to say, “..., if the characters ', \, //, or /\* occur in the sequence between the " delimiters, the behavior is undefined.” (Equivalent words exist in C++03 in §2.8, “Header names”.)