FloatingPoint

WORK IN PROGRESS

Floating Point Numbers, Infinities and NaNs

In preparation for resolving STDCXX-51 this page outlines the platform-specific details describing infinities and NaNs produced as the result of certain floating point calculations, usually triggered by invalid operands.

libc symbols

C99 specifies the following interface to infinities and NaNs:

7.12 Mathematics <math.h>

~~4~~ The macro INFINITY expands to a constant expression of type float representing positive or unsigned infinity, if available; else to a positive constant of type float that overflows at translation time.

~~5~~ The macro NAN is defined if and only if the implementation supports quiet NaNs for the float type. It expands to a constant expression of type float representing a quiet NaN.

7.12.11.2 The nan functions

~~1~~ Synopsis

```
#include <math.h>
double nan(const char *tagp);
float nanf(const char *tagp);
long double nanl(const char *tagp);
```

Description

~~2~~ The call nan("*n-char-sequence*") is equivalent to strtod("NAN(*n-charsequence*)", (char**) NULL); the call nan("") is equivalent to strtod ("NAN()", (char**) NULL). If tagp does not point to an *n-char sequence* or an empty string, the call is equivalent to strtod("NAN", (char**) NULL). Calls to nanf and nanl are equivalent to the corresponding calls to strtof and strtold.

Returns

~~3~~ The nan functions return a quiet NaN, if available, with content indicated through tagp. If the implementation does not support quiet NaNs, the functions return zero.

The table below lists the public symbols defined on each platform for Infinity, quiet NaN, and signaling NaN.

			libc symbols				
	AIX	HP-UX	IRIX	Linux	Solaris	Tru64 UNIX	Windows
header	<float.h>	<math.h>	N/A	<math.h>	<sunmath.h>	<float.h>	
Infinity	INFINITY, FLT_INF, DBL_INF, LDBL_INF	INFINITY	N/A	INFINITY	INFINITY, infinity()	FLT_INFINITY, DBL_INFINITY, LDBL_INFINITY	N/A
Quiet NaN	FLT_QNAN, DBL_QNAN, LDBL_QNAN [2]	nan(const char*) [1]	N/A	nan(const char*) [1]	quiet_nan(long) [2]	FLT_QNAN, DBL_QNAN, LDBL_QNAN	N/A
Signaling NaN	FLT_SNAN, DBL_SNAN, LDBL_SNAN		N/A	N/A	signaling_nan(long)	FLT_SNAN, DBL_SNAN, LDBL_SNAN	N/A

[1] Supports all signatures required by C99 and POSIX: float nanf(const char*), double nan(const char*), and long double nanl(const char*).

[2] Recent versions also support the C99 interface.

printf() formatting

				printf() formatting				
	AIX	HP-UX	IRIX	Linux	Solaris	Tru64 UNIX	Windows/x86	Windows/x64
Infinity	+/- INF	+/- inf	+/- inf	+/- inf	+/- Inf	+/- INF	+/- 1.#INF	+/- 1.#INF
Quiet NaN	+/- NaNQ	+/- nan	+/- nan0xxx	+/- nan	+/- NaN	NaNQ	~~-~~ 1.#IND	~~-~~ 1.#IND
Signaling NaN	+/- NaNS	+/- nan	+/- nan0xxx	+/- nan	+/- NaN	NaNS	+/- 1.#QNAN	+/- 1.#SNAN

numeric_limits<double> values/formatting

				numeric_limits<double> values/formatting				
	AIX	HP-UX	IRIX	Linux	Solaris	Tru64 UNIX	Windows /x86	Windows /x64
	XLC++ 9.0	aCC 3,6	MIPSpro	gcc	Sun C++	HP C++	MSVC	MSVC
`infinity()`	`INF`	`inf`	`inf`	`inf`	`Inf`	0	`1.#INF`	`1.#INF`
`quiet_NaN()`	`-NaNQ`	`nan`	`nan0xxx`	`nan`	`NaN`	0	`1.#QNAN`	`1.#QNAN`
`signaling_NaN()`	`-INF`	`nan`	`nan0xxx`	`nan`	`NaN`	0	`1.#QNAN`	`1.#SNAN`

Bit Patterns

The table below shows the bit patterns for Infinity, Quiet NaN, and Signaling NaN on each platform.

			IEEE 754 double precision bit patterns				
Number		Sign	Exponent	Fraction			
AIX/Power	Infinity	`0`	`0x7ff`	`0`			
Quiet NaN	`0`	`0x7ff`	`0x80000`				
Signaling NaN	`0`	`0x7ff`	`0x5555555500055555`				
HP-UX	Infinity	`0`	`0x7ff`	`0`			
Quiet NaN	`0`	`0x7ff`	`0x40000`				
Signaling NaN	`0`	`0x7ff`	`0x80000`				
IRIX/MIPS	Infinity						
Quiet NaN							
Signaling NaN							
Linux/x86	Infinity	`0`	`0x7ff`	`0`			
Quiet NaN[1]	`0`	`0x7ff`	`0x80000`				
Signaling NaN[1]	`0`	`0x7ff`	`0x40000`				
Solaris/SPARC	Infinity						
Quiet NaN							
Signaling NaN							
Solaris/x86	Infinity						
Quiet NaN[1]							
Signaling NaN[1]							
Tru64 UNIX /Alpha	Infinity	`0`	`0`	`0x7ff0000000000000`			
Quiet NaN	`0`	`0`	`0xfff8000000000000`				
Signaling NaN	`1`	`0x2aa`	`0x7ff5555500055555`				
Windows/x86	Infinity						
Quiet NaN							
Signaling NaN							

[1] Intel and AMD processors set the first fraction bit to 1 for Quiet NaNs and clear it for Signaling NaNs.