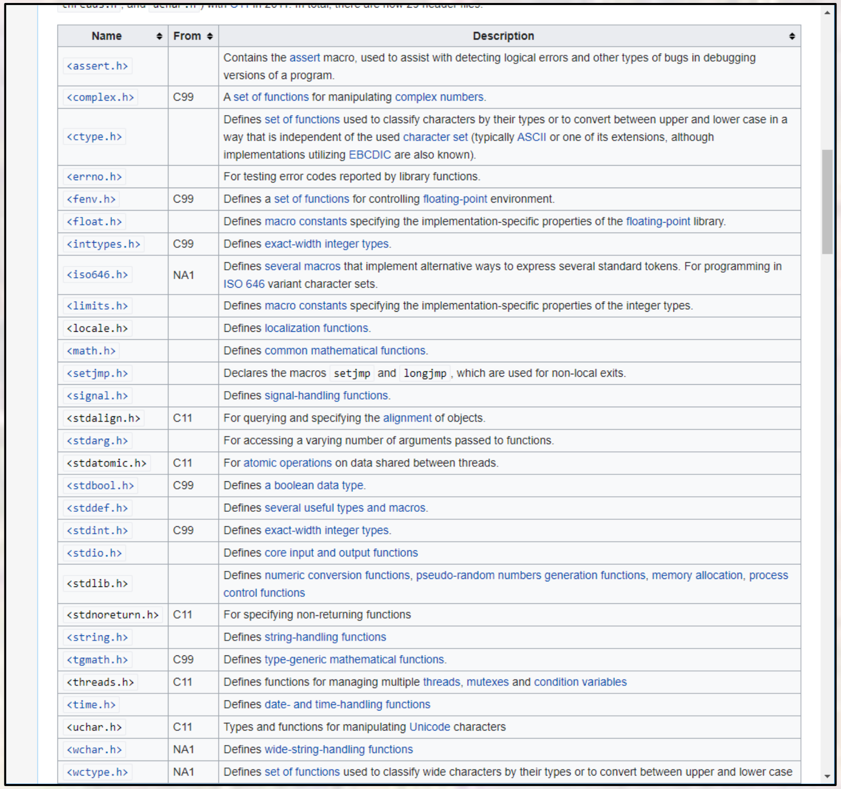
Standard C Functions

Last Updated 10/29/20

Standard Functions

- C standard functions
  - C function reference libraries
    - https://en.wikipedia.org/wiki/C_standard_library



Name	From	Description
<assert.h>		Contains the assert macro, used to assist with detecting logical errors and other types of bugs in debugging versions of a program.
<complex.h>	C99	A set of functions for manipulating complex numbers.
<ctype.h>		Defines set of functions used to classify characters by their types or to convert between upper and lower case in a way that is independent of the used character set (typically ASCII or one of its extensions, although implementations utilizing EBCDIC are also known).
<errno.h>		For testing error codes reported by library functions.
<fenv.h>	C99	Defines a set of functions for controlling floating-point environment.
<float.h>		Defines macro constants specifying the implementation-specific properties of the floating-point library.
<inttypes.h>	C99	Defines exact-width integer types.
<iso646.h>	NA1	Defines several macros that implement alternative ways to express several standard tokens. For programming in ISO 646 variant character sets.
<limits.h>		Defines macro constants specifying the implementation-specific properties of the integer types.
<locale.h>		Defines localization functions.
<math.h>		Defines common mathematical functions.
<setjmp.h>		Declares the macros setjmp and longjmp, which are used for non-local exits.
<signal.h>		Defines signal-handling functions.
<stdalign.h>	C11	For querying and specifying the alignment of objects.
<stdarg.h>		For accessing a varying number of arguments passed to functions.
<stdatomic.h>	C11	For atomic operations on data shared between threads.
<stdbool.h>	C99	Defines a boolean data type.
<stddef.h>		Defines several useful types and macros.
<stdint.h>	C99	Defines exact-width integer types.
<stdio.h>		Defines core input and output functions
<stdlib.h>		Defines numeric conversion functions, pseudo-random numbers generation functions, memory allocation, process control functions
<stdnoreturn.h>	C11	For specifying non-returning functions
<string.h>		Defines string-handling functions
<tgmath.h>	C99	Defines type-generic mathematical functions.
<threads.h>	C11	Defines functions for managing multiple threads, mutexes and condition variables
<time.h>		Defines date- and time-handling functions
<uchar.h>	C11	Types and functions for manipulating Unicode characters
<wchar.h>	NA1	Defines wide-string-handling functions
<wctype.h>	NA1	Defines set of functions used to classify wide characters by their types or to convert between upper and lower case

Standard Functions

- I/O Functions
  - C functions

    #include <stdio.h>
    - printf()
    - scanf()

Standard Functions

- Math Functions

```
#include <math.h>
```

Standard Functions

- Math Functions
  - Ceiling – ceil(x)
    ceilf, ceill
    - Smallest integer value ≥ the operand

    ceil(4.5) → 5
    ceil(-12.7) → -12

  - Floor – floor(x)
    floorf, floorl
    - Largest integer value ≤ the operand

    floor(4.5) → 4
    floor(-12.7) → -13

Standard Functions

- Math Functions
  - Truncate – trunc(x)
    truncf, truncl
    - Rounds toward 0

    trunc(4.5) → 4
    trunc(-12.7) → -12

  - Round – round(x)
    roundf, roundl
    - Rounds to nearest integer

    round(4.5) → 5
    round(-12.7) → -13

Standard Functions

- Math Functions
  - Power – pow(x,y)

    powf, powl
    - x to the power y

    pow(3.0, 4.0) → 81.0

    pow(3.4, 2.3) → 16.687893
  - Square Root – sqrt(x)

    sqrtf, sqrtlf
    - Square root of x

    sqrt(25) → 5.0

Standard Functions

- Math Functions
  - Absolute value – abs(x), fabs(x)
    labs, llabs, fabsf, fabsl

```
if(fabs(a-b) < 0.000001){
```