Returns Arc cosine of x; the return value is in the range 0 to π. A domain error occurs if x isn't between −1 and +1. *23.3*

---

acosh *Arc Hyperbolic Cosine (C99)* <math.h>

```
        double acosh(double x);
acoshf  float acoshf(float x);
acoshl  long double acoshl(long double x);
```

Returns Arc hyperbolic cosine of x; the return value is in the range 0 to +∞. A domain error occurs if x is less than 1. *23.4*

---

asctime *Convert Broken-Down Time to String* <time.h>

```
char *asctime(const struct tm *timeptr);
```

Returns A pointer to a null-terminated string of the form

```
Sun Jun  3 17:48:34 2007\n
```

constructed from the broken-down time in the structure pointed to by timeptr. *26.3*

---

asin *Arc Sine* <math.h>

```
        double asin(double x);
asinf   float asinf(float x);
asinl   long double asinl(long double x);
```

Returns Arc sine of x; the return value is in the range −π/2 to +π/2. A domain error occurs if x isn't between −1 and +1. *23.3*

---

asinh *Arc Hyperbolic Sine (C99)* <math.h>

```
        double asinh(double x);
asinhf  float asinhf(float x);
asinhl  long double asinhl(long double x);
```

Returns Arc hyperbolic sine of x. *23.4*

---

assert *Assert Truth of Expression* <assert.h>

void assert(*scalar* expression); *macro*

If the value of expression is nonzero, assert does nothing. If the value is zero, assert writes a message to stderr (specifying the text of expression, the name of the source file containing the assert, and the line number of the assert); it then terminates the program by calling abort. To disable assert, define the macro NDEBUG before including <assert.h>. *C99 changes:* The argument is allowed to have any scalar type; C89 specifies that the type is int. Also, C99 requires that the message written by assert include the name of the function in which the assert appears; C89 doesn't have this requirement. *24.1*

---

atan *Arc Tangent* <math.h>

```
        double atan(double x);
atanf   float atanf(float x);
```