log10 *Common Logarithm* <math.h>

```
double log10(double x);
float log10f(float x);
long double log10l(long double x);
```

Returns Logarithm of x to the base 10. A domain error occurs if x is negative. A range error may occur if x is zero. *23.3*

log1p *Natural Logarithm of 1 Plus Argument (C99)* <math.h>

```
double log1p(double x);
float log1pf(float x);
long double log1pl(long double x);
```

Returns Logarithm of 1 + x to the base *e*. A domain error occurs if x is less than –1. A range error may occur if x is equal to –1. *23.4*

log2 *Base-2 Logarithm (C99)* <math.h>

```
double log2(double x);
float log2f(float x);
long double log2l(long double x);
```

Returns Logarithm of x to the base 2. A domain error occurs if x is negative. A range error may occur if x is zero. *23.4*

logb *Radix-Independent Exponent (C99)* <math.h>

```
double logb(double x);
float logbf(float x);
long double logbl(long double x);
```

Returns $\log_r(|x|)$, where *r* is the radix of floating-point arithmetic (defined by the macro FLT_RADIX, which typically has the value 2). A domain error or range error may occur if x is zero. *23.4*

longjmp *Nonlocal Jump* <setjmp.h>

```
void longjmp(jmp_buf env, int val);
```

Restores the environment stored in env and returns from the call of setjmp that originally saved env. If val is nonzero, it will be setjmp's return value; if val is 0, setjmp returns 1. *24.4*

lrint *Round to Long Integer Using Current Direction (C99)* <math.h>

```
long int lrint(double x);
long int lrintf(float x);
long int lrintl(long double x);
```

Returns x rounded to the nearest integer using the current rounding direction. If the rounded value is outside the range of the long int type, the result is unspecified and a domain or range error may occur. *23.4*