Returns	(x) > (y). Unlike the > operator, `isgreater` doesn't raise the *invalid* floating-point exception if one or both of the arguments is a NaN.	*23.4*

isgreaterequal	*Test for Greater Than or Equal (C99)*	`<math.h>`
	`int isgreaterequal` *(real-floating* `x`, *real-floating* `y`)*;	*macro*
Returns	(x) >= (y). Unlike the >= operator, `isgreaterequal` doesn't raise the *invalid* floating-point exception if one or both of the arguments is a NaN.	*23.4*

isinf	*Test for Infinity (C99)*	`<math.h>`
	`int isinf` *(real-floating* `x`)*;	*macro*
Returns	A nonzero value if x is infinity (positive or negative) and zero otherwise.	*23.4*

isless	*Test for Less Than (C99)*	`<math.h>`
	`int isless` *(real-floating* `x`, *real-floating* `y`)*;	*macro*
Returns	(x) < (y). Unlike the < operator, `isless` doesn't raise the *invalid* floating-point exception if one or both of the arguments is a NaN.	*23.4*

islessequal	*Test for Less Than or Equal (C99)*	`<math.h>`
	`int islessequal` *(real-floating* `x`, *real-floating* `y`)*;	*macro*
Returns	(x) <= (y). Unlike the <= operator, `islessequal` doesn't raise the *invalid* floating-point exception if one or both of the arguments is a NaN.	*23.4*

islessgreater	*Test for Less Than or Greater Than (C99)*	`<math.h>`
	`int islessgreater` *(real-floating* `x`, *real-floating* `y`)*;	*macro*
Returns	(x) < (y) || (x) > (y). Unlike this expression, `islessgreater` doesn't raise the *invalid* floating-point exception if one or both of the arguments is a NaN; also, x and y are evaluated only once.	*23.4*

islower	*Test for Lower-Case Letter*	`<ctype.h>`
	`int islower(int c);`	
Returns	A nonzero value if c is a lower-case letter and zero otherwise.	*23.5*

isnan	*Test for NaN (C99)*	`<math.h>`
	`int isnan` *(real-floating* `x`)*;	*macro*
Returns	A nonzero value if x is a NaN value and zero otherwise.	*23.4*

isnormal	*Test for Normal Number (C99)*	`<math.h>`
	`int isnormal` *(real-floating* `x`)*;	*macro*
Returns	A nonzero value if x has a normal value (not zero, subnormal, infinite, or NaN) and zero otherwise.	*23.4*

isprint	*Test for Printing Character*	`<ctype.h>`
	`int isprint(int c);`	