

Manual pages for the C standard library, the C POSIX library, and the CS50 Library for those less comfortable.

Search

frequently used in CS50

cs50.h

`get_char` - prompt a user for a `char`
`get_double` - prompt a user for a `double`
`get_float` - prompt a user for a `float`
`get_int` - prompt a user for an `int`
`get_long` - prompt a user for an `long`
`get_string` - prompt a user for a `string`

ctype.h

`isalnum` - check whether a character is alphanumeric
`isalpha` - check whether a character is alphabetical
`isblank` - check whether a character is blank (i.e., a space or tab)
`isdigit` - check whether a character is a digit
`islower` - check whether a character is lowercase
`ispunct` - check whether a character is punctuation
`isspace` - check whether a character is whitespace (e.g., a newline, space, or tab)
`isupper` - check whether a character is uppercase
`tolower` - convert a `char` to lowercase
`toupper` - convert a `char` to uppercase

math.h

`ceil` - calculate the ceiling of a number
`floor` - calculate the floor of a number
`log2` - calculate the base-2 logarithm of a number
`pow` - raise a number to a power

`round` - round a number to the nearest integer

`sqrt` - calculate the square root of a number

stdio.h

`fclose` - close a file

`fopen` - open a file

`fprintf` - print to a file

`fread` - read bytes from a file

`fscanf` - get input from a file

`fwrite` - write bytes to a file

`printf` - print to the screen

`scanf` - get input from a user

`sprintf` - print to a string

stdlib.h

`atof` - convert a `string` to a `float`

`atoi` - convert a `string` to an `int`

`atol` - convert a `string` to a `long`

`free` - free dynamically allocated memory

`malloc` - allocate memory dynamically

`random` - generate a pseudorandom number

`realloc` - reallocate memory dynamically

`srandom` - seed pseudorandom number generation

string.h

`strcasestr` - locate a substring

`strcmp` - compare two strings

`strcpy` - copy a string

`strlen` - calculate the length of a string

`strstr` - locate a substring

strings.h

`strcasecmp` - compare two strings ignoring case

[time.h](#)

[time](#) - get time in seconds