

**NAME**

`cacosh`, `cacoshf`, `cacoshl` – complex arc hyperbolic cosine

**SYNOPSIS**

```
#include <complex.h>
```

```
double complex cacosh(double complex z);
```

```
float complex cacoshf(float complex z);
```

```
long double complex cacoshl(long double complex z);
```

Link with `-lm`.

**DESCRIPTION**

These functions calculate the complex arc hyperbolic cosine of  $z$ . If  $y = \text{cacosh}(z)$ , then  $z = \text{ccosh}(y)$ . The imaginary part of  $y$  is chosen in the interval  $[-\pi, \pi]$ . The real part of  $y$  is chosen nonnegative.

One has:

$$\text{cacosh}(z) = 2 * \text{clog}(\text{csqrt}((z + 1) / 2) + \text{csqrt}((z - 1) / 2))$$
**VERSIONS**

These functions first appeared in `glibc` in version 2.1.

**ATTRIBUTES**

For an explanation of the terms used in this section, see [attributes\(7\)](#).

Interface	Attribute	Value
<code>cacosh()</code> , <code>cacoshf()</code> , <code>cacoshl()</code>	Thread safety	MT-Safe

**CONFORMING TO**

C99, POSIX.1-2001, POSIX.1-2008.

**EXAMPLE**

```
/* Link with "-lm" */
#include <complex.h>
#include <stdlib.h>
#include <unistd.h>
#include <stdio.h>

int
main(int argc, char *argv[])
{
    double complex z, c, f;

    if (argc != 3) {
        fprintf(stderr, "Usage: %s <real> <imag>\n", argv[0]);
        exit(EXIT_FAILURE);
    }

    z = atof(argv[1]) + atof(argv[2]) * I;
    c = cacosh(z);
    printf("cacosh() = %6.3f %6.3f*i\n", creal(c), cimag(c));

    f = 2 * clog(csqrt((z + 1)/2) + csqrt((z - 1)/2));
    printf("formula = %6.3f %6.3f*i\n", creal(f), cimag(f));

    exit(EXIT_SUCCESS);
}
```

**SEE ALSO**

[acosh\(3\)](#), [cabs\(3\)](#), [ccosh\(3\)](#), [cimag\(3\)](#), [complex\(7\)](#)

**COLOPHON**

This page is part of release 4.16 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.