

NAME

ctime, *localtime*, *gmtime* – convert date and time to ASCII

SYNOPSIS

```
char *ctime(tvec)
int tvec[2];

[from Fortran]
double precision ctime
... = ctime(dummy)

int *localtime(tvec)
int tvec[2];

int *gmtime(tvec)
int tvec[2];
```

DESCRIPTION

Ctime converts a time in the vector *tvec* such as returned by *time* (II) into ASCII and returns a pointer to a character string in the form

```
Sun Sep 16 01:03:52 1973\n\0
```

All the fields have constant width.

The *localtime* and *gmtime* entries return pointers to integer vectors containing the broken-down time. *Localtime* corrects for the time zone and possible daylight savings time; *gmtime* converts directly to GMT, which is the time UNIX uses. The value is a pointer to an array whose components are

| | |
|---|---------------------------------------|
| 0 | seconds |
| 1 | minutes |
| 2 | hours |
| 3 | day of the month (1-31) |
| 4 | month (0-11) |
| 5 | year – 1900 |
| 6 | day of the week (Sunday = 0) |
| 7 | day of the year (0-365) |
| 8 | Daylight Saving Time flag if non-zero |

The external variable *timezone* contains the difference, in seconds, between GMT and local standard time (in EST, is 5*60*60); the external variable *daylight* is non-zero iff the standard U.S.A. Daylight Savings Time conversion should be applied. The program knows about the peculiarities of this conversion in 1974 and 1975; if necessary, a table for these years can be extended.

A routine named *ctime* is also available from Fortran. Actually it more resembles the *time* (II) system entry in that it returns the number of seconds since the epoch 0000 GMT Jan. 1, 1970 (as a floating-point number).

SEE ALSO

time(II)

BUGS