

# time.h

**time.h** é um arquivo cabeçalho que fornece protótipos para funções, macros e definição de tipos da biblioteca padrão da linguagem de programação C para manipulação de datas e horários<sup>[1]</sup> de modo padrão.

## Variáveis

size\_t

clock\_t Variável para guardar o tempo de processador.

time\_t Variável para guardar o tempo de calendário, e geralmente corresponde ao número de segundos decorridos desde as 00 horas do dia 1 de Janeiro de 1970 UTC.

struct tm Estrutura para guardar data e hora. É composta pelos elementos:

```
struct tm
{
    int tm_sec;           /* Seconds. [0-60] (1 leap second) */
    int tm_min;           /* Minutes. [0-59] */
    int tm_hour;          /* Hours. [0-23] */
    int tm_mday;          /* Day. [1-31] */
    int tm_mon;           /* Month. [0-11] */
    int tm_year;          /* Year - 1900. */
    int tm_wday;          /* Day of week. [0-6] */
    int tm_yday;          /* Days in year. [0-365] */
    int tm_isdst;         /* DST. [-1/0/1] */
}
```

## Macros

NULL

CLOCKS\_PER\_SEC

## Funções

char \*asctime(const struct tm\* tmptr)

Converte tm para uma string no formato "Www Mmm dd hh:mm:ss yyyy", onde Www é o dia da semana, Mmm o mês em letras, dd o dia do mês, hh:mm:ss a hora, e yyyy o ano. A string é seguida por uma nova linha e um caracter nulo (NULL), formada por um total de 26 caracteres. A string gerada é alocada estaticamente e compartilhada pelas funções ctime e asctime. Cada vez que uma dessas funções é chamada o conteúdo da string é substituído.

clock\_t clock(void)

Retorna o tempo decorrido desde que o programa iniciou a execução.

### Biblioteca padrão do C

- [assert.h](#)
- [complex.h](#)
- [ctype.h](#)
- [errno.h](#)
- [fenv.h](#)
- [float.h](#)
- [inttypes.h](#)
- [iso646.h](#)
- [limits.h](#)
- [locale.h](#)
- [math.h](#)
- [setjmp.h](#)
- [signal.h](#)
- [stdarg.h](#)
- [stdbool.h](#)
- [stddef.h](#)
- [stdint.h](#)
- [stdio.h](#)
- [stdlib.h](#)
- [string.h](#)
- [tgmath.h](#)
- [time.h](#)
- [wchar.h](#)
- [wctype.h](#)

**char\* ctime(const time\_t\* *timer*)**

Converte o valor de time\_t para uma string com um formato igual ao de asctime. Essa string gerada é alocada estaticamente e compartilhada pelas funções ctime e asctime. Cada vez que uma dessas funções é chamada, ou usada, o conteúdo da string é subescrito. asctime também usa internamente o arquivo temporário usado por gmtime e localtime como valor de retorno, então uma chamada dessa função vai alterar esse buffer (ou arquivo temporário).

**double difftime(time\_t *timer2*, time\_t *timer1*)**

Retorna em segundos a diferença entre dois espaços de tempo.

**struct tm\* gmtime(const time\_t\* *timer*)**

Converte um valor time\_t para uma estrutura tm como o relógio UTC. Essa estrutura é alocada estaticamente e usada também pelas funções gmtime, localtime e ctime. Essa estrutura tem seu valor alterado cada vez que uma dessas funções é chamada.

**struct tm\* gmtime\_r(const time\_t\* *timer*, struct tm\* *result*)**

Converte um valor time\_t para uma estrutura tm igual ao UTC. O tempo é armazenado na estrutura tm referenciada pelo *resultado*. Essa função é uma versão segura da função gmtime.

**struct tm\* localtime(const time\_t\* *timer*)**

Converte um valor time\_t para uma estrutura tm, igual ao relógio local. Essa estrutura é alocada estaticamente e usada também pelas funções gmtime, localtime e ctime. Essa estrutura tem seu valor alterado cada vez que uma dessas funções é chamada.

**time\_t mktime(struct tm\* *ptm*)**

Converte tm para valor de tempo time\_t. Checa os membros da estrutura *tm* passada como parâmetro *ptm* ajustando os valores se esses não estão no reange, ou tamanho certo ou estão incompletos ou errados e então traduz a estrutura para um valor time\_t então retornado. O valor original do *dia da semana*, membro do *tm\_mdia* não é checado enquanto *tm\_mes* e *tm\_ano* estão determinados. Caso haja erro, o valor -1 é retornado.

**time\_t time(time\_t\* *timer*)**

Pega a data atual, em segundos de acordo com o relógio Unix, a partir do relógio do sistema. Guarda esse valor no *timer*. Se o *timer* é igual a *null*, o valor não é armazenado, mas esse valor é continuamente retornado pela função.

**size\_t strftime(char\* *s*, size\_t *n*, const char\* *format*, const struct tm\* *tptr*)**

Formata tm numa string data/hora cujo resultado é colocado no array *s*.

String Formatação	Significado
%a	Abreviatura do dia da semana
%A	Dia da semana
%b	Mês abreviado
%B	Mês
%c	Data e hora
%d	Dia do mês
%H	Hora no formato 24H (00-23)
%I	Hora no formato 12H (01-12)
%j	Dia do ano
%m	Mês no formato numérico
%M	Minuto
%p	Indicador AM ou PM
%S	Segundos
%U	Dia da semana com o primeiro Domingo como o primeiro dia da primeira semana (00-53)
%w	Dia da semana numérico com Domingo como 0 (0-6)
%W	Dia da semana com a primeira Segunda Feira como o primeiro dia da primeira semana (00-53)
%x	Representação da data
%X	Representação da hora
%y	Ano a dois dígitos
%Y	Ano a quatro dígitos
%Z	Fuso horário
%%	Símbolo %

```
char * strptime(const char* buf, const char* format, struct tm* tptr)  
time_t timegm(struct tm *brokentime)
```

## Exemplo

O código abaixo imprime numa tela do terminal uma mensagem com o dia, mês, ano e horário.

```
#include <stdio.h>  
#include <time.h>  
  
int main()  
{  
    const time_t timer = time(NULL);  
    printf("ctime is %s\n", ctime(&timer));  
}
```

```
    return 0;  
}
```

## Referências

---

- «C Time Library» (<http://www.cplusplus.com/reference/clibrary/ctime/>). The C++ Resources Network. Consultado em 15 de março de 2010

## Ligações externas

---

- «Calendar Time - The GNU C Library» ([http://www.gnu.org/software/libc/manual/html\\_node/Calendar-Time.html](http://www.gnu.org/software/libc/manual/html_node/Calendar-Time.html)) (em inglês) line feed character character in |título= at position 14 ([ajuda](#))
- «gmtime». *The Open Group Base Specifications* (<http://www.opengroup.org/onlinepubs/009695399/functions/gmtime.html>). [S.l.: s.n.] 9 de dezembro de 2008

---

Obtida de "<https://pt.wikipedia.org/w/index.php?title=Time.h&oldid=64401155>"

■