

linux时间函数 PDF转换可能丢失图片或格式，建议阅读原文  
[https://www.100test.com/kao\\_ti2020/645/2021\\_2022\\_linux\\_E6\\_97\\_B6\\_E9\\_97\\_c103\\_645688.htm](https://www.100test.com/kao_ti2020/645/2021_2022_linux_E6_97_B6_E9_97_c103_645688.htm) linux时间函数我们在编程中可能会经常用到时间，比如取得系统的时间（获取系统的年、月、日、时、分、秒，星期等），或者是隔一段时间去做某事，那么我们就用到一些时间函数。linux下存储时间常见的有两种存储方式，一个是从1970年到现在经过了多少秒，一个是用一个结构来分别存储年月日时分秒的。time\_t这种类型就是用来存储从1970年到现在经过了多少秒，要想更精确一点，可以用结构struct timeval，它精确到微妙。struct timeval { long tv\_sec; /\*秒\*/ long tv\_usec; /\*微妙\*/ }; 而直接存储年月日的是一个结构：struct tm { int tm\_sec; /\*秒，正常范围0-59，但允许至61\*/ int tm\_min; /\*分钟，0-59\*/ int tm\_hour; /\*小时，0-23\*/ int tm\_mday; /\*日，即一个月中的第几天，1-31\*/ int tm\_mon; /\*月，从一月算起，0-11\*/ int tm\_year; /\*年，从1900至今已经多少年\*/ int tm\_wday; /\*星期，一周中的第几天，从星期日算起，0-6\*/ int tm\_yday; /\*从今年1月1日到目前的天数，范围0-365\*/ int tm\_isdst; /\*日光节约时间的旗标\*/ }; 需要特别注意的是，年份是从1900年起至今多少年，而不是直接存储如2008年，月份从0开始的，0表示一月，星期也是从0开始的，0表示星期日，1表示星期一。下面介绍一下我们常用的时间函数：  
#include char \*asctime ( const struct tm\* timeptr ); 将结构中的信息转换为真实世界的时间，以字符串的形式显示char \*ctime ( const time\_t \*timep ); 将timep转换为真是世界的时间，以字符串显示，它和asctime不同就在于传入的参数形式不一样double difftime

( time\_t time1 , time\_t time2 ) ; 返回两个时间相差的秒数  
int gettimeofday ( struct timeval \*tv , struct timezone \*tz ) ; 返回当前距离1970年的秒数和微妙数 , 后面的tz是时区 , 一般不用  
struct tm\* gmtime ( const time\_t \*timep ) ; 将time\_t表示的时间转换为没有经过时区转换的UTC时间 , 是一个struct tm结构指针  
struct tm\* localtime ( const time\_t \*timep ) ; 和gmtime类似 , 但是它是经过时区转换的时间。  
time\_t mktime ( struct tm\* timeptr ) ; 将struct tm结构的时间转换为从1970年至今的秒数  
time\_t time ( time\_t \*t ) ; 取得从1970年1月1日至今的秒数。  
上面是简单的介绍 , 下面通过实战来看看这些函数的用法 :  
下载 : `gettime1.c /*gettime1.c*/ #include int main ( ) { time_t timep ; time (`  
100Test 下载频道开通 , 各类考试题目直接下载。  
详细请访问 [www.100test.com](http://www.100test.com)