

## Re: using mktime()

**Source:** [http://coding.derkeiler.com/Archive/C\\_CPP/comp.lang.c/2005-02/2892.html](http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2005-02/2892.html)

---

**From:** Michael Mair (*Michael.Mair\_at\_invalid.invalid*)

**Date:** 02/21/05

Date: Mon, 21 Feb 2005 15:30:40 +0100

Al Bowers wrote:

> *Dave Thompson wrote:*

>

>>>>

>>>

>>> *No. What the Standard says is that function mktime will bring  
>>> all values into range. For example the range for struct tm member*

>>>

>>

>>

>> *If the call is successful, yes. It (definitely) won't be if the  
>> requested time is not representable in time\_t, and it's not clear if  
>> mktime() is allowed to fail in other cases that the implementor  
>> decides are "too hard" -- the comments in the Olson public-domain  
>> implementation imply to me that this might have happened.*

>>

>>

>>

> *Yes, the Standard does not specify anything on the resulting values of  
> the struct tm*

> *members should and when an implement of function mktime decides that  
> time is*

> *not representable. The values may be changed or unchanged, in part or  
> in total.*

>

>>> *tm\_sec is 0-59. If tm\_sec has the value of say -69 the function  
>>> mktime will bring tm\_sec into range by subtracting 1 from tm\_min.*

>>>

>>

>>

>> *Actually tm\_sec is 0-60 to allow for (positive) leap seconds, which  
>> are rarely if ever implemented. That is, leap seconds actually happen  
>> (for now, there has been discussion of eliminating them) but (most?) C  
>> implementations (and systems) just treat them as transient errors.  
>> The only people I've heard of actually using them are the ones for  
>> whom they were designed -- astronomers and space navigators, and their  
>> only contact to most ordinary people, GPS.*