MCR10133
Add prototype for C `time()` function to `time.h`
Revision 1.0

Eric Swenson

June 17, 2023

1 Problem Description

The "C Users's Guide", HH07, on page 4-202, describes the runtime library function `time()` as being present, yet it does not appear in any header file, including the obvious `time.h` header file.

The issue is reported in this ticket: https://multics-trac.swenson.org/ticket/187.

2 Proposed Fix

The proposed fix is simply to add a prototype for the `time()` function to `time.h`. The documentation in HH07 says that this function returns the time. It further says that you can call it with a pointer to a `long`, where the time value will be stored. If you pass NULL for this pointer, the implication is that the time will be returned as the return value of the function.

Experimenting shows that the return value is always zeroed and that the only way to get the time returned is by passing a pointer to a `long`.

So either of these prototypes:

```
extern long time();
```

```
extern int time();
```

may be acceptable. But since ISO C declares it as:

```
time_t time(time_t *tloc);
```

but K&R C doesn't allow the parameters to be declared in the prototype, probably the best prototype would be:

```
extern long time();
```

even though the return value appears always to be 0.

A `compare_ascii` output of `time.h` showing the proposed fix is below:

```
cpa >sl3p>cc>incl>time.h ==

Inserted in B:
B18      extern long time();
Preceding:
A18      extern long timezone;

Comparison finished: 1 difference, 1 line.
```

3 Documentation

No changes to documentation are necessary.

4 Test

The following test program is used to check the compilation and execution of a program that calls the `time` function.

```
#include <stdio.h>
#include <time.h>

main()
{
         long tloc;
         long retval;

         tloc = 77;
         retval = 42;

         retval = time(&tloc);
/*       retval = time(NULL); */
         printf("time: tloc=%ld\n", tloc);
         printf("time: retval=%ld\n", retval);
}
```

Compiling and running the test program shows that the declaration works and the function works (to the extent mentioned above).

5 Version History

2023-06-17	1.0	Initial version of the MCR.