```c
 1: // $Id: mktimestamp.c,v 1.2 2014-02-20 15:32:43-08 - - $
 2:
 3: //
 4: // Open several files and write into them.
 5: // Then use utime(2) to force a time stamp onto them.
 6: //
 7:
 8: #include <errno.h>
 9: #include <libgen.h>
10: #include <stdio.h>
11: #include <stdlib.h>
12: #include <string.h>
13: #include <sys/types.h>
14: #include <time.h>
15: #include <utime.h>
16:
17: char *execname = NULL;
18: int exit_status = EXIT_SUCCESS;
19:
20: void maketimefile (time_t when) {
21:    char filename[64];
22:    sprintf (filename, "timestamp.%016lX", when);
23:    FILE *file = fopen (filename, "w");
24:    if (file == NULL) {
25:       fprintf (stderr, "%s: %s: %s\n",
26:                execname, filename, strerror (errno));
27:       exit_status = EXIT_FAILURE;
28:       return;
29:    }
30:    printf ("fopen (%s): OK\n", filename);
31:    char buffer[64];
32:    strftime (buffer, sizeof buffer, "%c %Z", localtime (&when));
33:    fprintf (file, "%s\n", buffer);
34:    strftime (buffer, sizeof buffer, "%c %Z", gmtime (&when));
35:    fprintf (file, "%s\n", buffer);
36:    fclose (file);
37:    struct utimbuf utimbuf;
38:    utimbuf.actime = when;
39:    utimbuf.modtime = when;
40:    utime (filename, &utimbuf);
41: }
42:
43: int main (int argc, char **argv) {
44:    (void) argc; // warning: unused parameter 'argc'
45:    execname = basename (argv[0]);
46:    time_t now = time (NULL);
47:    const time_t DAYS = 24 * 60 * 60;
48:    maketimefile (-0x80000000L);
49:    maketimefile (0);
50:    maketimefile (now - 200 * DAYS);
51:    maketimefile (now);
52:    maketimefile (now + 200 * DAYS);
53:    maketimefile (0x7FFFFFFF);
54:    maketimefile (0xFFFFFFFFL);
55:    maketimefile (0xFFFFFFFFFFL);
56:    return exit_status;
57: }
```

```
58:
59:
60: //TEST// rm ./timestamp.* >mktimestamp.lis 2>&1
61: //TEST// ./mktimestamp >>mktimestamp.lis 2>&1
62: //TEST// ls -goatr timestamp.* >>mktimestamp.lis 2>&1
63: //TEST// grep . timestamp.* >>mktimestamp.lis 2>&1
64: //TEST// l8stat.perl *.perl *.c timestamp.* >>mktimestamp.lis 2>&1
65: //TEST// mkpspdf mktimestamp.ps mktimestamp.c* *.perl mktimestamp.lis
66:
```

```
    1: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting mktimestamp.c
    2: mktimestamp.c:
    3:      $Id: mktimestamp.c,v 1.2 2014-02-20 15:32:43-08 - - $
    4: gcc -g -O0 -Wall -Wextra -std=gnu11 mktimestamp.c -o mktimestamp -lglut
-lGLU -lGL -lX11 -lm
    5: rm -f mktimestamp.o
    6: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished mktimestamp.c
```

```perl
 1: #!/usr/bin/perl
 2: # $Id: l8stat.perl,v 1.1 2012-02-29 19:04:02-08 - - $
 3: use POSIX qw(strftime);
 4: $0 =~ s|.*/||;
 5: $days180 = 180 * 24 * 3600;
 6: for $file (@ARGV ? @ARGV : ".") {
 7:    ($_, $_, $mode, $_, $_, $_, $_, $size, $_, $mtime, $_, $_, $_)
 8:          = lstat $file;
 9:    print STDERR "$0: $file: $!\n" and next unless defined $size;
10:    $fmt = $mtime < $^T - $days180 || $^T + $days180 < $mtime
11:        ? "%b %e  %Y" : "%b %e %R";
12:    printf "%06o %9d %s %s%s\n",
13:            $mode, $size, (strftime $fmt, localtime $mtime), $file,
14:            defined ($link = readlink $file) ? " -> $link" : "";
15: }
```

```
 1: fopen (timestamp.FFFFFFFF80000000): OK
 2: fopen (timestamp.0000000000000000): OK
 3: fopen (timestamp.000000005272E346): OK
 4: fopen (timestamp.00000000537A8F46): OK
 5: fopen (timestamp.0000000054823B46): OK
 6: fopen (timestamp.000000007FFFFFFF): OK
 7: fopen (timestamp.00000000FFFFFFFF): OK
 8: fopen (timestamp.000000FFFFFFFFFF): OK
 9: -rw-------. 1 58 Dec 13  1901 timestamp.FFFFFFFF80000000
10: -rw-------. 1 60 Dec 31  1969 timestamp.000000FFFFFFFFFF
11: -rw-------. 1 58 Dec 31  1969 timestamp.00000000FFFFFFFF
12: -rw-------. 1 58 Dec 31  1969 timestamp.0000000000000000
13: -rw-------. 1 58 Oct 31  2013 timestamp.000000005272E346
14: -rw-------. 1 58 May 19 16:09 timestamp.00000000537A8F46
15: -rw-------. 1 58 Dec  5  2014 timestamp.0000000054823B46
16: -rw-------. 1 58 Jan 18  2038 timestamp.000000007FFFFFFF
17: timestamp.0000000000000000:Wed Dec 31 16:00:00 1969 PST
18: timestamp.0000000000000000:Thu Jan  1 00:00:00 1970 GMT
19: timestamp.000000005272E346:Thu Oct 31 16:09:58 2013 PDT
20: timestamp.000000005272E346:Thu Oct 31 23:09:58 2013 GMT
21: timestamp.00000000537A8F46:Mon May 19 16:09:58 2014 PDT
22: timestamp.00000000537A8F46:Mon May 19 23:09:58 2014 GMT
23: timestamp.0000000054823B46:Fri Dec  5 15:09:58 2014 PST
24: timestamp.0000000054823B46:Fri Dec  5 23:09:58 2014 GMT
25: timestamp.000000007FFFFFFF:Mon Jan 18 19:14:07 2038 PST
26: timestamp.000000007FFFFFFF:Tue Jan 19 03:14:07 2038 GMT
27: timestamp.00000000FFFFFFFF:Sat Feb  6 22:28:15 2106 PST
28: timestamp.00000000FFFFFFFF:Sun Feb  7 06:28:15 2106 GMT
29: timestamp.000000FFFFFFFFFF:Sun Feb 19 16:36:15 36812 PST
30: timestamp.000000FFFFFFFFFF:Mon Feb 20 00:36:15 36812 GMT
31: timestamp.FFFFFFFF80000000:Fri Dec 13 12:45:52 1901 PST
32: timestamp.FFFFFFFF80000000:Fri Dec 13 20:45:52 1901 GMT
33: 100700       586 Oct 18  2013 l8stat.perl
34: 100600      1040 May 19 16:09 futuretimes.c
35: 100600      1875 May 19 16:09 mktimestamp.c
36: 100600       381 Mar  5 19:17 now.c
37: 100600      1601 Feb 20 15:29 showlink.c
38: 100600        58 Dec 31  1969 timestamp.0000000000000000
39: 100600        58 Oct 31  2013 timestamp.000000005272E346
40: 100600        58 May 19 16:09 timestamp.00000000537A8F46
41: 100600        58 Dec  5  2014 timestamp.0000000054823B46
42: 100600        58 Jan 18  2038 timestamp.000000007FFFFFFF
43: 100600        58 Dec 31  1969 timestamp.00000000FFFFFFFF
44: 100600        60 Dec 31  1969 timestamp.000000FFFFFFFFFF
45: 100600        58 Dec 13  1901 timestamp.FFFFFFFF80000000
```