

```

/*                                     tab:8
 *
 * truth-table.c - example of multi-bit logic computation in C
 *
 * "Copyright (c) 2016 by Steven S. Lumetta."
 *
 * Permission to use, copy, modify, and distribute this software and its
 * documentation for any purpose, without fee, and without written agreement is
 * hereby granted, provided that the above copyright notice and the following
 * two paragraphs appear in all copies of this software.
 *
 * IN NO EVENT SHALL THE AUTHOR OR THE UNIVERSITY OF ILLINOIS BE LIABLE TO
 * ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL
 * DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION,
 * EVEN IF THE AUTHOR AND/OR THE UNIVERSITY OF ILLINOIS HAS BEEN ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * THE AUTHOR AND THE UNIVERSITY OF ILLINOIS SPECIFICALLY DISCLAIM ANY
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
 * MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE
 * PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND NEITHER THE AUTHOR NOR
 * THE UNIVERSITY OF ILLINOIS HAS ANY OBLIGATION TO PROVIDE MAINTENANCE,
 * SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS."
 *
 * Author:           Steve Lumetta
 * Version:          2
 * Creation Date:    2 September 2016
 * Filename:         truth-table.c
 * History:
 *   SL              1          2 September 2016
 *                   First written.
 *   SL              2          3 September 2016
 *                   Removed operators not used in ECE120.
 */

#include <stdint.h>
#include <stdio.h>

int
main ()
{
    /*
     * The input variables A, B, and C are 8-bit unsigned values.
     * We use each bit to represent a possible combination of the
     * three variables. Bit 7 of each is set to a 1, for example.
     * Bit 4 of A is set to 1, while bits 4 of B and C are set to 0.
     * In this way, we cover all entries of the truth table for
     * F(A,B,C).
     */
    uint8_t A = 0xF0; /* input variable A          */
    uint8_t B = 0xCC; /* input variable B          */
    uint8_t C = 0xAA; /* input variable C          */
    uint8_t F;       /* the function F           */
    int32_t i;       /* truth table row iteration variable */

    /*
     * Compute all possible values of function F using one statement.
     *   F(A,B,C) = (A+B)(A'+C')
     */
    F = ((A | B) & ((~A) | (~C)));

```

```

/* Print a truth table for F. */
printf ("A B C | F\n");
printf ("-----+---\n");
for (i = 0; 8 > i; i = i + 1) {
    printf ("%c %c %c | %c\n",
            '0' + (0 != (i & 4)), /* A      */
            '0' + (0 != (i & 2)), /* B      */
            '0' + (0 != (i & 1)), /* C      */
            '0' + (0 != (F & (1 << i)))); /* F(A,B,C) */
}

/* Return success. */
return 0;
}

```