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 != (F & (1 << i)))); /* F(A,B,C) */
}