Aplicații ale algebrei booleene

- C++:

```
.................
    // We need some TRUE/FALSE masks to convert CCA_BOOL values to something
    // that can be "anded" with the db field masks.
    int BOOL_TRUE_MASK = 0xFFFFFFFF;
    int BOOL_FALSE_MASK = 0x00000000;
    int BOOL_MASK;

    // Set the last connect time.
    m_DbInMemory.device.tlast[deviceIndex] = theDevice->GetLastConnectTime();

    // Set the in service flag.  In the database and on the display it is
    // Not In Service so it will be the opposite of the inservice? return.
    if ( !theDevice->IsInService() )
                BOOL_MASK = BOOL_TRUE_MASK;
   else
                BOOL_MASK = BOOL_FALSE_MASK;

    m_DbInMemory.device.service[deviceIndex] = ( NIS_DEVICE & BOOL_MASK );
.....................
byte NIS_GATE =                 {x, x, busy, close, open, x, off, x}
1. Testam on/off ➔ Mask_OFF=  {0, 0,   0,    0,     0,  0,  1 , 0}
      if (NIS_GATE & Mask_OFF) {0, 0,   0,    0,     0,  0, 1/0, 0}
         TRUE - off=1
         FALSE- off=0
1. Daca e on -> e busy?
```

Exemplu

```
! volatile byte NIS_GATE =      {x, x, busy, close, open, x, off, x}
1. Testam on/off ➔ Mask_OFF=  {0, 0,   0,    0,     0,  0,  1 , 0}
      if (NIS_GATE & Mask_OFF) {0, 0,   0,    0,     0,  0, 1/0, 0}
          TRUE – off=1
          FALSE- off=0
2. Daca e on -> e busy?
 Mask_BUSY=  {
3. Daca nu e busy -> e open?
 Mask_OPEN=  {


COD:
if (NIS_GATE & Mask_OFF)        //on?off   (1)
{         //true -> este OFF
          //eroare
          …
}
else
{         //false – este ON (alimentata)
          if (NIS_GATE & Mask_BUSY)    //busy?  (2)
          {
          …
          }
          else
          {

          }
}
```