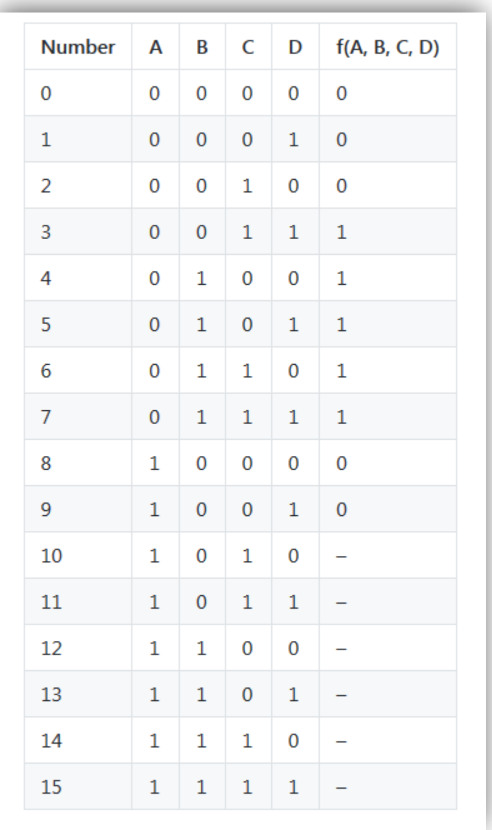
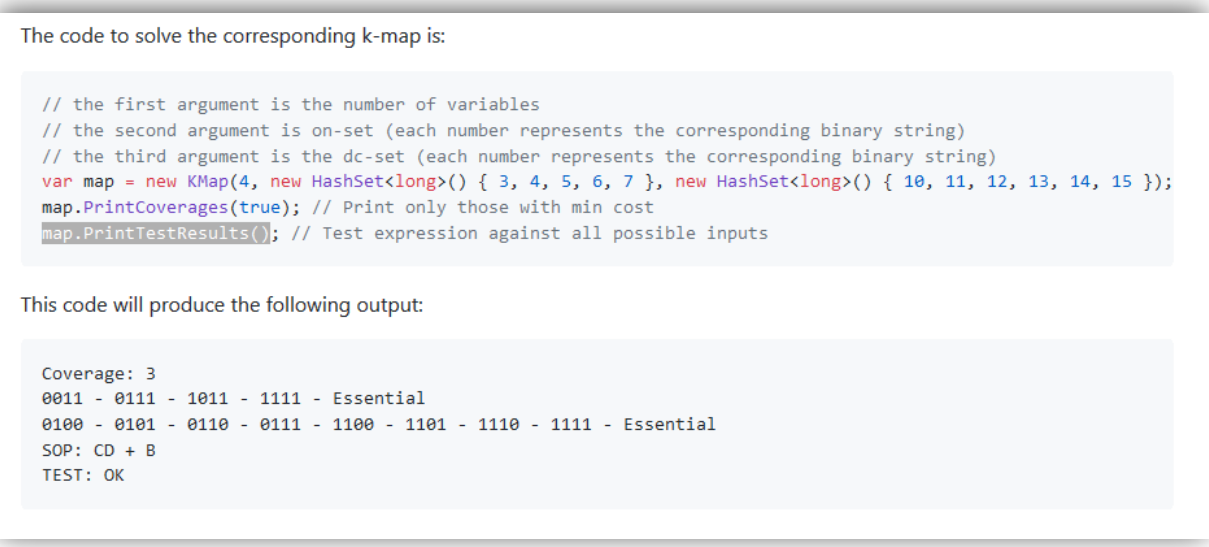
Karnaugh

https://github.com/hkproj/karno

Przykład:



Number	A	B	C	D	f(A, B, C, D)
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	–
11	1	0	1	1	–
12	1	1	0	0	–
13	1	1	0	1	–
14	1	1	1	0	–
15	1	1	1	1	–



The code to solve the corresponding k-map is:

```
// the first argument is the number of variables
// the second argument is on-set (each number represents the corresponding binary string)
// the third argument is the dc-set (each number represents the corresponding binary string)
var map = new KMap(4, new HashSet<long>() { 3, 4, 5, 6, 7 }, new HashSet<long>() { 10, 11, 12, 13, 14, 15 });
map.PrintCoverages(true); // Print only those with min cost
map.PrintTestResults(); // Test expression against all possible inputs
```

This code will produce the following output:

```
Coverage: 3
0011 - 0111 - 1011 - 1111 - Essential
0100 - 0101 - 0110 - 0111 - 1100 - 1101 - 1110 - 1111 - Essential
SOP: CD + B
TEST: OK
```