Booleova algebra

Pojem a podstata Booleovy algebry

George Boole zavedl v polovině 19. století zvláštní druh algebry, který našel uplatnění až koncem třicátých let 20. století při studiu spínacích obvodů.

Booleovu algebru budeme chápat jako nauku o operacích na **množině B**, která obsahuje **dvě logické konstanty 0** a **I** a dále **logické proměnné,** které označujeme např. x, y, z, …, a, b, c, … Booleova algebra používá tento **základní soubor operací:**

- **logický součin,** který se běžně označuje operátorem $\wedge$. V technické praxi se vžilo označování pomocí tečky (.), popř. s vynecháním tečky;
- **logický součet,** který se běžně označuje $\vee$. V dalším textu budeme užívat +.
- **negace,** která se značí značkou $\neg$ před operátorem. V dalším textu budeme užívat též pruh nad operandem.

Logická proměnná může nabývat pouze dvou hodnot 0 a I. Booleova algebra není algebrou čísel, ale stavů.

Logické funkce budeme zapisovat pravdivostní tabulkou. Pro n proměnných, z nichž každá může nabývat dvou stavů, dostaneme $k = 2^n$ kombinací. Pro n proměnných existuje 2^k funkcí.

Definice logické funkce

V dalším textu budeme předpokládat pouze logickou funkci úplně zadanou, tj. takovou funkci, u které známe její hodnotu 0 nebo I pro všechny možné kombinace hodnot proměnných. Pro takovou funkci lze sestavit **pravdivostní tabulku.**

Příklad: funkce tří proměnných f(a, e, o). Funkce je zadaná pravdivostní tabulkou (TAB. 1):

TAB. 1

o	e	a	f
0	0	0	I
0	0	I	0
0	I	0	I
0	I	I	I
I	0	0	0
I	0	I	0
I	I	0	I
I	I	I	I

Libovolnou funkci f(a, b, c, …) lze zapsat ve dvou základních tvarech zvaných **základní součtový tvar** (též úplná normální disjunktivní forma) a **základní součinový tvar** (též úplná konjunktivní normální forma).

Funkci zapsanou v **základním součtovém tvaru** dostaneme jako součet základních součinů přímých nebo negovaných proměnných. Zapisujeme pouze základní součiny **(mintermy)** u těch kombinací přímých nebo negovaných proměnných, u kterých má funkce hodnotu I; v mintermu zapisujeme proměnné, které nabývají v příslušné kombinaci hodnoty I jako přímé, a proměnné, které nabývají hodnoty 0 jako negované.

Funkci zapsanou v **základním součinovém tvaru** dostaneme jako součin základních součtů přímých nebo negovaných proměnných. Zapisujeme pouze základní součty **(maxtermy)** u těch kombinací přímých nebo negovaných proměnných, u kterých má funkce hodnotu 0; v maxtermu zapisujeme proměnné, které nabývají v příslušné kombinaci hodnoty 0 jako přímé, a proměnné, které nabývají hodnoty I jako negované.

V praxi se používají mnohem častěji mintermy než maxtermy. V následující tabulce (TAB. 2) je zápis všech mintermů a maxtermů pro logické funkce tří proměnných.

TAB. 2

o	**e**	**a**	**Mintermy**	**Maxtermy**
0	0	0	$\bar{a}\bar{e}\bar{o}$	$a+e+o$
0	0	I	$a\bar{e}\bar{o}$	$\bar{a}+e+o$
0	I	0	$\bar{a}e\bar{o}$	$a+\bar{e}+o$
0	I	I	$ae\bar{o}$	$\bar{a}+\bar{e}+o$
I	0	0	$\bar{a}\bar{e}o$	$a+e+\bar{o}$
I	0	I	$a\bar{e}o$	$\bar{a}+e+\bar{o}$
I	I	0	$\bar{a}eo$	$a+\bar{e}+\bar{o}$
I	I	I	aeo	$\bar{a}+\bar{e}+\bar{o}$

Pokračování příkladu: Pomocí mintermů zapíšeme funkci (viz TAB. 1):

$$f = \bar{a}\bar{e}\bar{o} + \bar{a}e\bar{o} + ae\bar{o} + \bar{a}eo + aeo$$

Funkce jedné proměnné

Ve shodě s posledním odstavcem první části pro jednu proměnnou platí, že n = 1, takže kombinací je $k = 2^1 = 2$ a možných funkcí tedy $2^2 = 4$ (TAB. 3).

TAB. 3

a	f_0	f_1	f_2	f_3
0	0	0	I	I
I	0	I	0	I

Dostali jsme čtyři funkce:

$f_0 = 0$ a $f_3 = I$ konstanty

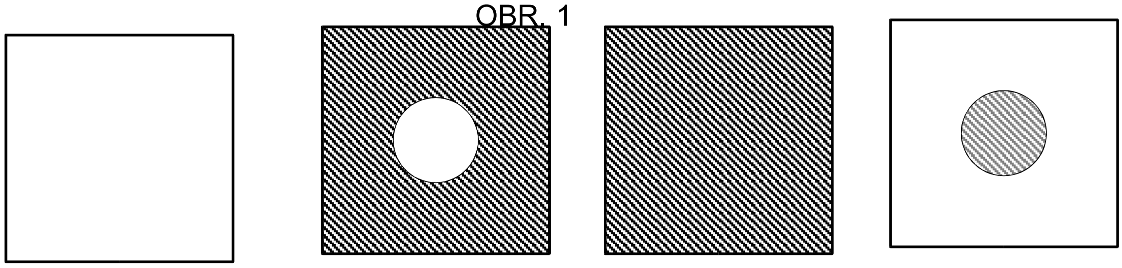
$f_1 = a$ proměnná sama

$f_2 = \bar{a}$ negace proměnné.

Praktický význam má pouze negace proměnné.

Jednotlivé funkce si můžeme graficky znázornit pomocí **Vennových diagramů** (OBR. 1). Vyšrafovaná část představuje I a nevyšrafovaná 0.

OBR. 1



$f_0 = 0$ $f_3 = \mathrm{I}$ $f_1 = a$ $f_2 = 0$

Pro jednu proměnnou platí tyto **zákony:**

- **zákon dvojité negace**: $\neg(\bar{a}) = a$
- **zákony vyloučeného třetího**: $a + \bar{a} = \mathrm{I}$
  $a \,.\, \bar{a} = 0$
- **zákony neutrálnosti nuly a jedničky**: $a + 0 = a$
  $a \,.\, \mathrm{I} = a$
- **zákony agresivity nuly a jedničky:** $a \,.\, 0 = 0$
  $a + \mathrm{I} = \mathrm{I}$

Funkce dvou proměnných

Dvě proměnné mají čtyři možné kombinace, takže funkcí dvou proměnných je celkem 16. Jsou dány pravdivostní tabulkou (TAB. 4).

TAB. 4

e	a	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	I	I	I	I	I	I	I	I
0	I	0	0	0	0	I	I	I	I	0	0	0	0	I	I	I	I
I	0	0	0	I	I	0	0	I	I	0	0	I	I	0	0	I	I
I	I	0	I	0	I	0	I	0	I	0	I	0	I	0	I	0	I

Jistou funkci (např. f_5) vyjádříme z tabulky v základním součtovém tvaru

$$f_5 = a\bar{e} + ae$$

Po zjednodušení funkcí najdeme:

Funkce jedné proměnné

$f_0 = 0$	a	$f_{15} = \mathrm{I}$	konstanty
$f_5 = a$	a	$f_3 = e$	proměnná sama
$f_{10} = \bar{a}$	a	$f_{12} = \bar{e}$	negace proměnné

Dvě funkce odpovídající základním operátorům

$f_7 = a + e$ logický součet

Logický součet

Funkce logický součet též **logický součet ve významu slučovacím** a také **disjunkce.** Logický součet se běžně zapisuje znakem $\vee$ a v běžné řeči se vyznačuje slovem **„nebo“.** Logický součet je pravdivý, když alespoň jeden operand je pravdivý. Tato funkce má anglickou zkratku **OR.**

$f_1 = a \,.\, e$ logický součin

Logický součin

Funkce logický součin též **konjunkce.** Logický součin se běžně zapisuje znakem $\wedge$ a v běžné řeči se vyznačuje slovem **„i“.** Logický součin je pravdivý, když všechny operandy jsou pravdivé. Tato funkce má anglickou zkratku **AND.**

Osm nových funkcí

$f_6 = a\bar{e} + \bar{a}e$ nonekvivalence

Nonekvivalence

Funkce nonekvivalence **neboli součet ve významu vylučovacím,** též **součet modulo dvě,** či **exkluzivní nebo,** popř. **kontravalence,** se běžně zapisuje znakem $\oplus$ nebo **>—<,** v běžné řeči se čte **„buď … anebo“**. Nonekvivalence je negací ekvivalence a anglickou zkratku má **XOR.**

$f_9 = ae + \bar{a}\bar{e}$ ekvivalence

Ekvivalence

Funkce ekvivalence, se běžně zapisuje znakem $\equiv$ či $\Leftrightarrow$, v logice se vyjadřuje **„tehdy a jedině tehdy, když“** nebo **„právě když“.** Operátor ekvivalence vyjadřuje **nutnou a dostačující podmínku.** Ekvivalence je pravdivá jen tehdy, mají-li oba operandy stejnou pravdivostní hodnotu.

$f_8 = \bar{a}\bar{e} = \neg(a + e)$ negace logického součtu

Negace logického součtu

Funkce negace logického součtu neboli **„ani jeden není“** se označuje též jako **Piercova funkce.** Má anglickou zkratku **NOR.**

$f_{14} = \bar{a} + \bar{e} = \neg(a\ .\ e)$ negace logického součinu

Negace logického součinu

Funkce negace logického součinu neboli **aspoň jeden není** se označuje též jako **Shefferova funkce,** popř. jako **exkluze**. Má anglickou zkratku **NAND.** V běžné řeči se vyjadřuje jako **„ne zároveň … a“.** Jako operátor se většinou užívá **|**.

$f_{13} = \bar{e} + a$ Implikace

Implikace

Funkce implikace se vyjadřuje jako **„jestliže …, pak …“.** Jako operátor se většinou užívá $\Rightarrow$. Implikace vyjadřuje **formální vztah dostačující podmínky.** Implikace je nepravdivá, jen když první operand je pravdivý a druhý nepravdivý. Pozor: implikace se nedá obrátit, zde záleží na pořadí operandů. Takže z tabulky podle TAB. 4 zapisujeme **e** $\Rightarrow$ **a.**

$f_2 = \bar{a}e$

$f_4 = a\bar{e}$

$f_{11} = \bar{a} + e$

Pro dvě proměnné můžeme formulovat další **zákony:**

- **zákony komutativní** $a + e = e + a$
  $a\ .\ e = e\ .\ a$
- **zákony absorpce** $a + a.e = a$
  $a\ .\ (a + b) = a$
- **zákony absorpce negace** $a + \bar{a}.e = a + e$
  $a\ .\ (\bar{a} + e) = a\ .\ e$
- **De Morganovy zákony** $\neg(a + e) = \bar{a}.\bar{e}$
  $\neg(a\ .\ e) = \bar{a} +.\bar{e}$

Algebraické zjednodušování logických funkcí

Při algebraickém zjednodušování logických funkcí lze užít výše uvedených zákonů a dalších zákonů:

- **zákony asociativní** $(a + e) + o = a + (e + o)$
  $(a\ .\ e)\ .\ o = a\ .\ (e\ .\ o)$
- **zákony distributivní** $(a + e)\ .\ o = a.o + e.o$
  $a.e + o = (a + o)\ .\ (e + o)$

Priorita logických operací

1. závorky
2. negace
3. logický součin
4. logický součet

Zopakujme funkci z našeho příkladu a zjednodušme ji:

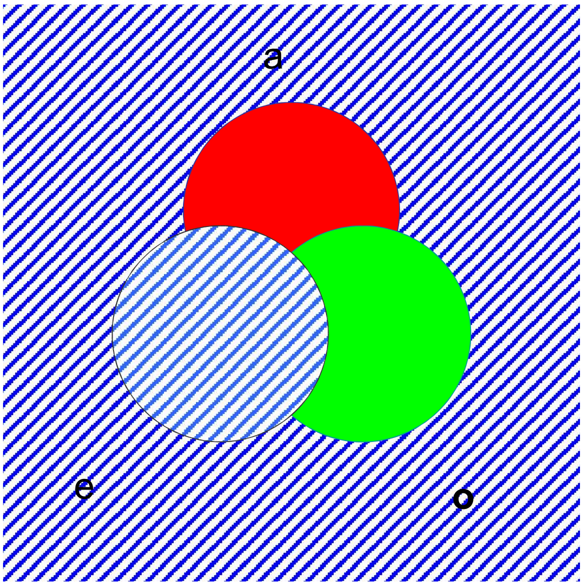
$$f = \bar{a}\bar{e}\bar{o} + \bar{a}e\bar{o} + ae\bar{o} + \bar{a}eo + aeo$$

$$f = \bar{a}\bar{e}\bar{o} + e\,.(\bar{a}\bar{o} + a\bar{o} + \bar{a}o + ao) = \bar{a}\bar{e}\bar{o} + e\,.[\bar{a}\,.\,(\bar{o} + o) + a\,.\,(o + \bar{o})] =$$

$$= \bar{a}\bar{e}\bar{o} + e\,.[\bar{a}\,.\,(1) + a\,.\,(1)] = \bar{a}\bar{e}\bar{o} + e\,.(\bar{a} + a) = \bar{a}\bar{e}\bar{o} + e\,.(1) = = \bar{a}\bar{e}\bar{o} + e$$

Výsledek zjednodušení můžeme zakreslit do Vennova diagramu (OBR. 2):

OBR. 2



Literatura:

[1] Jáneš, V. a kol.: Technické vybavení počítačů, SPN, Praha 1982
[2] Blatný, J. a kol.: Číslicové počítače, SNTL, Praha 1982
[3] Bernard, J.-M., Od logických obvodů k mikroprocesorům I, SNTL, Praha 1984
[4] Anzenbacher A., Úvod do filozofie, SPN, Praha 1990
[5] Lukasová A., Formální logika v umělé inteligenci, Computer Press, Brno 2003