

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ В
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота №1

з дисципліни
«Дискретна математика»

Виконала:
студентка групи КН-115
Дзямба Аліна
Викладач:
Мельникова Н.І.

Львів – 2019 р.

Моделювання основних логічних операцій

Мета роботи: Ознайомитись на практиці із основними поняттями математичної логіки, навчитись будувати складні висловлювання за допомогою логічних операцій та знаходити їхні істинностні значення таблицями істинності, використовувати закони алгебри логіки, освоїти методи доведень.

Варіант №7

Додаток 1

1. Формалізувати речення. Багато непорозумінь між урядами України та Польщі, але ні Україна, ні Польща не втратили економічної співпраці.
2. Побудувати таблицю істинності для висловлювань:
 $((\bar{x} \Leftrightarrow \bar{y}) \Leftrightarrow ((z \Rightarrow (x \vee y)) \Rightarrow \bar{z}));$
3. Побудовою таблиць істинності вияснити, чи висловлювання є тавтологією або протиріччям: $((p \vee q) \wedge (q \wedge r)) \rightarrow (p \vee r)$
4. За означенням без побудови таблиць істинності та виконання еквівалентних перетворень перевірити, чи є тавтологією висловлювання: $((p \rightarrow q) \wedge (p \rightarrow q)) \rightarrow (\bar{p} \rightarrow r);$
5. Довести, що формули еквівалентні: $p \leftrightarrow (q \vee r)$ та $p \wedge (q \rightarrow r).$

Розв'язання:

1. x — уряд України
 y — уряд Польщі
 P — багато непорозумінь
 R — втратити економічну співпрацю
 $P(x, y) \wedge \overline{R(x, y)}$

2. $((\overset{\textcircled{1}}{\bar{x}} \Leftrightarrow \overset{\textcircled{5}}{\bar{y}}) \Leftrightarrow ((\overset{\textcircled{3}}{z} \Rightarrow (\overset{\textcircled{2}}{x} \vee \overset{\textcircled{4}}{y})) \Rightarrow \bar{z}))$

x	y	z	\bar{x}	\bar{y}	\bar{z}	①	②	③	④	⑤
F	F	F	T	T	T	T	F	T	T	T

F	F	T	T	T	F	T	F	F	T	T
F	T	F	T	F	T	F	T	T	T	F
F	T	T	T	F	F	F	T	T	F	T
T	F	F	F	T	T	F	T	T	T	F
T	F	T	F	T	F	F	T	T	F	T
T	T	F	F	F	T	T	T	T	T	T
T	T	T	F	F	F	T	T	T	F	F

$$3. \quad \overset{\textcircled{1}}{\overline{(p \vee q)}} \wedge \overset{\textcircled{2}}{\overline{(q \wedge r)}} \rightarrow (p \vee r)$$

r	q	p	$p \vee q$	$\overline{(p \vee q)}$	$q \wedge r$	$\overline{(q \wedge r)}$	$p \vee r$	①	②
F	F	F	F	T	F	T	F	T	F
F	F	T	T	F	F	T	T	F	T
F	T	F	T	F	F	T	F	F	T
F	T	T	T	F	F	T	T	F	T
T	F	F	F	T	F	T	T	T	T
T	F	T	T	F	F	T	T	F	T
T	T	F	T	F	T	F	T	F	T
T	T	T	T	F	T	F	T	F	T

Висловлювання $\overline{((p \vee q) \wedge (q \wedge r))} \rightarrow (p \vee r)$ не є ні тавтологією, ні протиріччям.

$$4. \quad ((p \rightarrow q) \wedge (p \rightarrow q)) \rightarrow (\bar{p} \rightarrow r)$$

Припустимо, що

$$((p \rightarrow q) \wedge (p \rightarrow q)) = T$$

$$(\bar{p} \rightarrow r) = F$$

Тоді припустимо, що

$$\bar{p} = T$$

$$q = F$$

Підставимо $q = F$ у рівність

$$((p \rightarrow F) \wedge (p \rightarrow F)) = T$$

Значення Т отримаємо, якщо

$$(p \rightarrow F) = T \quad \text{і} \quad (p \rightarrow F) = T$$

Рівності виконуються при $p = F$

Отже, дане висловлювання не є тавтологією.

$$5. \quad p \leftrightarrow (q \vee r)$$

$$p \wedge (q \rightarrow r)$$

Перевіримо, чи $(p \leftrightarrow (q \vee r)) \leftrightarrow (p \wedge (q \rightarrow r))$

$$\begin{matrix} \textcircled{2} & \textcircled{1} & \textcircled{5} & \textcircled{4} & \textcircled{3} \\ (p \leftrightarrow (q \vee r)) & \leftrightarrow & (p \wedge (q \rightarrow r)) \end{matrix}$$

p	q	r	①	②	③	④	⑤
F	F	F	F	T	T	F	F
F	F	T	T	F	T	F	T
F	T	F	T	F	F	F	T
F	T	T	T	F	T	F	T
T	F	F	F	F	T	T	F
T	F	T	T	T	T	T	T
T	T	F	T	T	F	F	F
T	T	T	T	T	T	T	T

Отже, формули $(p \leftrightarrow (q \vee r))$ та $(p \wedge (q \rightarrow r))$ не еквівалентні.

Додаток 2

$$((\bar{x} \Leftrightarrow \bar{y}) \Leftrightarrow ((z \Rightarrow (x \vee y)) \Rightarrow \bar{z}))$$

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <iostream>
4  using namespace std;
5
6  int main()
7  {
8      int x, y, z;
9
10     cout << "Enter 'x'\n";
11     cin >> x;
12     cout << "Enter 'y'\n";
13     cin >> y;
14     cout << "Enter 'z'\n";
15     cin >> z;
16
17     if ((x == 0) && (y == 0) && (z == 0))
18     {
19         cout << ("_x = 1, _y = 1, _z = 1\n");
20         cout << ("_x ~ _y = 1\n");
21         cout << ("x v y = 0\n");
22         cout << ("z - (x v y) = 1\n");
23         cout << ("(z - (x v y)) - _z = 1\n");
24         cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1\n");
25     }
26     else if ((x == 0) && (y == 0) && (z == 1))
27     {
28         cout << ("_x = 1, _y = 1, _z = 0\n");
29         cout << ("_x ~ _y = 1\n");
30         cout << ("x v y = 0\n");
31         cout << ("z - (x v y) = 0\n");
32         cout << ("(z - (x v y)) - _z = 1\n");
33         cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1\n");

```

```

34     }
35     else if ((x == 0) && (y == 1) && (z == 0))
36     {
37         cout << ("_x = 1, _y = 0, _z = 1\n");
38         cout << ("_x ~ _y = 0\n");
39         cout << ("x v y = 1\n");
40         cout << ("z - (x v y) = 1\n");
41         cout << ("(z - (x v y)) - _z = 1\n");
42         cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 0\n");
43     }
44     else if ((x == 0) && (y == 1) && (z == 1))
45     {
46         cout << ("_x = 1, _y = 0, _z = 0\n");
47         cout << ("_x ~ _y = 0\n");
48         cout << ("x v y = 1\n");
49         cout << ("z - (x v y) = 1\n");
50         cout << ("(z - (x v y)) - _z = 0\n");
51         cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1\n");
52     }
53     else if ((x == 1) && (y == 0) && (z == 0))
54     {
55         cout << ("_x = 0, _y = 1, _z = 1\n");
56         cout << ("_x ~ _y = 0\n");
57         cout << ("x v y = 1\n");
58         cout << ("z - (x v y) = 1\n");
59         cout << ("(z - (x v y)) - _z = 1\n");
60         cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 0\n");
61     }
62     else if ((x == 1) && (y == 0) && (z == 1))
63     {
64         cout << ("_x = 0, _y = 1, _z = 0\n");
65         cout << ("_x ~ _y = 0\n");
66         cout << ("x v y = 1\n");

```

```

67 cout << ("z - (x v y) = 1\n");
68 cout << ("(z - (x v y)) - _z = 0\n");
69 cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1\n");
70 }
71 else if ((x == 1) && (y == 1) && (z == 0))
72 {
73     cout << ("_x = 0, _y = 0, _z = 1\n");
74     cout << ("_x ~ _y = 1\n");
75     cout << ("x v y = 1\n");
76     cout << ("z - (x v y) = 1\n");
77     cout << ("(z - (x v y)) - _z = 1\n");
78     cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1\n");
79 }
80 else if ((x == 1) && (y == 1) && (z == 1))
81 {
82     cout << ("_x = 0, _y = 0, _z = 0\n");
83     cout << ("_x ~ _y = 1\n");
84     cout << ("x v y = 1\n");
85     cout << ("z - (x v y) = 1\n");
86     cout << ("(z - (x v y)) - _z = 0\n");
87     cout << ("( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 0\n");
88 }
89 else
90 {
91     cout << ("Error\n");
92 }
93 return 0;
94 }

```

Результат:

```

Enter 'x'
1
Enter 'y'
0
Enter 'z'
1
_x = 0, _y = 1, _z = 0
_x ~ _y = 0
x v y = 1
z - (x v y) = 1
(z - (x v y)) - _z = 0
( _x ~ _y ) ~ (( z - ( x v y )) - _z ) = 1

```

При вводі хоча б однієї змінної, відмінної від 0 чи 1, в результаті отримуємо "Error":

```

Enter 'x'
0
Enter 'y'
0
Enter 'z'
9
Error

```

Висновок: Я ознайомилась на практиці із основними поняттями математичної логіки, навчилась будувати складні висловлювання за допомогою логічних операцій та знаходити їхні істинні значення таблицями істинності, використовувати закони алгебри логіки, освоїла методи доведень.