

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота №3

з дисципліни «Дискретна математика»

Виконав:

студент групи КН-112

Тиський Святослав

Викладач:

Мельникова Н.І.

Львів-2019

Варіант №14

Лабораторна робота № 3.

Тема: Побудова матриці бінарного відношення

Мета роботи: набуття практичних вмінь та навичок при побудові матриць бінарних відношень та визначені їх типів.

Завдання №1: Чи є вірною рівність $A \times (B \cap C \cup D) = (A \times B) \cap (A \times C) \cup (A \times D)$?

Завдання №2: Знайти матрицю відношення $R \subset 2^A \times 2^B$:

$$R = \{(x, y) \mid x \subset A \& y \subset B \& |y| \leq x\}, \text{ де } A = \{1, 3\}, B = \{2, 4\}.$$

Завдання №3: Зобразити відношення графічно:

$$\alpha = \{(x, y) \mid (x, y) \in R^2 \& |6 - 3y| = x\}, \text{ де } R - \text{множина дійсних чисел.}$$

Завдання №4: Маємо бінарне відношення $R \subset A \times A$, де $A = \{a, b, c, d, e\}$, яке

задане своєю матрицею: $A(R) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$. Перевірити чи є дане

відношення рефлексивним, симетричним, транзитивним, антисиметричним?

Завдання №5: Визначити множину (якщо це можливо), на якій дане

відношення є: а) функціональним; б) бієктивним:

$$\alpha = \{(x, y) \mid (x, y) \in R^2 \& |x| + |y| = 4\}.$$

Розв'язок завдання №1:

$$(x, y) \in A \times (B \cap C \cup D) = (x \in A) \& (y \in B \& y \in C \mid y \in D) =$$

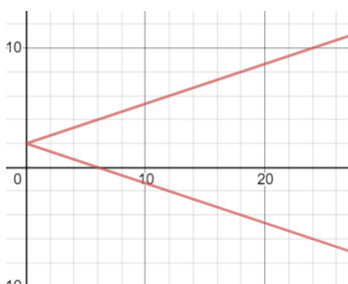
$$(x \in A \& y \in B \& x \in A \& y \in C \mid x \in A \& y \in D) =$$

$$(x, y) \in ((A \times B) \cap (A \times C) \cup (A \times D)) = (A \times B) \cap (A \times C) \cup (A \times D)$$

Розв'язок завдання №2:

	\emptyset	$\{2\}$	$\{4\}$	$\{2,4\}$
\emptyset	0	0	0	0
$\{1\}$	1	1	1	0
$\{3\}$	1	1	1	1

Розв'язок завдання №3:



$$\alpha = \{(x, y) \mid (x, y) \in R^2 \& |6 - 3y| = x\}, \text{ де } R - \text{множина дійсних чисел.}$$

Розв'язок завдання №4:

Матриця: $A(R)$ має вигляд $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}$.

Дане відношення є:

- Рефлексивне (вздовж головної діагоналі одиниці).
- Симетричне так як $\theta_{12} = \theta_{21}$, $\theta_{23} = \theta_{32}$ і т.д.
- Не транзитивне тому, що $\theta_{33} = 1$, $\theta_{44} = 1$, $\theta_{34} \neq 1$.

Розв'язок завдання №5:

а) Відношення $\alpha = \{(x, y) | (x, y) \in R^2 \& |x| + |y| = 4\}$ є функціональне якщо $\alpha = \{(x, y) | x \in [-4; 4] \& (y \in [0; 4] \vee y \in [-4; 0])\}$.

б) Відношення $\alpha = \{(x, y) | (x, y) \in R^2 \& |x| + |y| = 4\}$ є бієктивне якщо $\alpha = \{(x, y) | (x \in [0; 4] \& (y \in [0; 4] \vee y \in [-4; 0])) \vee x \in [-4; 0] \& (y \in [0; 4] \vee y \in [-4; 0])\}$.

Додаток №2:

Написати програму, яка знаходить матрицю бінарного відношення $\rho \subseteq A \times B$, заданого на двох числових множинах. Реалізувати введення цих множин, та виведення на екран матриці відношення. Перевірити програмно якого типу є задане відношення. Навести різні варіанти тестових прикладів.

Розв'язок додатка №2:

```
1  #include <iostream>
2  #include <locale>
3  using namespace std;
4
5  int main()
6  {
7      setlocale(LC_ALL, "Ukrainian");
8
9      //////////////////////////////////////////////////
10
11     int Pot1, Pot2;
12
13     cout << "Введіть потужність 1 множини: ";
14     cin >> Pot1;
15     cout << "Введіть потужність 2 множини: ";
16     cin >> Pot2;
17
18     //////////////////////////////////////////////////
19
20     float* arr1 = new float[Pot1];           //створюємо динамічний масив для 1 множини
21     float* arr2 = new float[Pot1];           //створюємо динамічний масив для 2 множини
22
23     cout << "Введіть 1 множину: ";
24     for (int i=0; i < Pot1; i++) {           //
25         cin >> arr1[i];                     //вводимо 1 множину
26     }                                         //
27
28     cout << "Введіть 2 множину: ";
29     for (int i=0; i < Pot1; i++) {           //
30         cin >> arr2[i];                     //вводимо 2 множину
31     }                                         //
32
33     //////////////////////////////////////////////////
34
35     int** arr3 = new int*[Pot1];             //створюємо динамічний масив вказівників(рядки)
36     //
37     for (int i = 0; i < Pot1; i++) {         //
```

```

37     for (int i = 0; i < Pot1; i++) {           //
38         arr3[i] = new int [Pot2];             //створюємо стовпці
39     }                                           //
40
41     for (int i = 0; i < Pot1; i++) {           //
42         for (int j = 0; j < Pot2; j++) {       //
43             if (arr1[i] > 2 * arr2[j]) {       //створюється
44                 arr3[i][j] = 1;               //бінарний
45             }                                 //масив
46             else {                             //що задовольняє
47                 arr3[i][j] = 0;               //умову
48             }                                 //a>2b
49         }                                     //
50     }                                         //
51     //////////////////////////////////////////
52
53     delete [] arr1;                           //видаляємо з пам'яті 1 масив
54     delete [] arr2;                           //видаляємо з пам'яті 2 масив
55
56     //////////////////////////////////////////
57
58     for (int i = 0; i < Pot1; i++) {           //
59         for (int j = 0; j < Pot2; j++) {       //виводимо
60             cout << arr3[i][j]<<"\t";         //бінарний
61         }                                     //масив
62         cout << endl;                         //
63     }                                         //
64
65     //////////////////////////////////////////
66
67     int check = 0;                            //перевірка
68     for (int i = 0; i < Pot1; i++) {           //на
69         if (arr3[i][i] == 1) {                 //рефлексивність
70             check++;                           //якщо на гол діагоналі 1 то check++
71         }                                     //
72     }                                         //

```

```

73
74     if (check == Pot1) {                      //
75         cout << "Рефлексивна ";               //якщо вся гол діагональ 1 то матриця транзитивна
76     }                                         //
77
78     else if (check == 0) {                    //
79         cout << "Антирефлексивна ";           //якщо вся гол діагональ 0 то матриця антитранзитивна
80     }                                         //
81
82     else {                                    //
83         cout << "Нерефлексивна ";             //якщо є 1 і 0 то матриця нетранзитивна
84     }                                         //
85
86     //////////////////////////////////////////
87
88     int check0=0;                            //
89     int check1=0;                            //
90     for (int i = 0; i < Pot1; i++) {           //перевірка
91         for (int j = 0; j < Pot2; j++) {       //
92             if (arr3[i][j]==arr3[j][i] && i!=j){ //
93                 check0++;                     //
94             }                                 //симетричність
95             else {                             //
96                 check1++;                     //матриці
97             }                                 //
98         }                                     //
99     }                                         //
100
101     if (check0 == Pot1*Pot2-Pot1) {           //
102         cout << "Симетрична ";                 //
103     }
104     else if (check1 == Pot1*Pot2-Pot1) {       //
105         cout << "Антисиметрична ";             //
106     }
107     else{
108         cout << "Несиметрична ";               //
109     }

```

```

108     cout << "Несиметрична ";
109 }
110
111 //////////////////////////////////////////////////
112
113 int tranz = 0, tranz1 = 1; //
114 for (int i = 0; i < Pot1; i++) { //
115     for (int j = 0; j < Pot2; j++) { //
116         if (arr3[i][j] == 1) { //перевірка на транзитивність
117             for (int z = 0; z < Pot1; z++) { //якщо arr3[i][j]=1
118                 if (arr3[j][z] == 1) { //починається цикл
119                     if (arr3[i][z] == 1) { //якщо знаходиться ще 1 одиниця від j
120                         tranz = 1; //шукаємо 3 одиницю
121                     } //якщо 3 одиниця є тоді транзитивність є
122                     else { //в іншому випадку немає
123                         tranz1 = 0; //
124                     } //
125                 } //
126             } //
127         } //
128     } //
129 } //
130
131 if (tranz * tranz1 == 1) { //
132     cout << "Транзитивна "; //Якщо є транзитивність
133 } //виводимо що матриця транзитивна
134 else { //в іншому випадку перевіряємо на анти транзитивність
135     int antitransz = 0;
136     for (int i = 0; i < Pot1; i++) { //
137         for (int j = 0; j < Pot2; j++) { //
138             if (arr3[i][j] == 1) { //перевірка на антитранзитивність
139                 for (int z = 0; z < Pot1; z++) { //якщо arr3[i][j]=1
140                     if (arr3[j][z] == 1) { //починається цикл
141                         if (arr3[i][z] == 0) { //якщо знаходиться ще 1 одиниця від j
142                             antitransz = 1; //шукаємо 3 нуль
143                         } //якщо 3 нуль є тоді антитранзитивність є
144                     } else { //в іншому випадку немає
145                         antitransz = 0; //
146                     } //
147                 } //
148             } //
149         } //
150     } //
151 } //
152
153 if (antitransz == 1) { //якщо матриця антитранзитивна
154     cout << "Антитранзитивна "; //виводимо що вона антитранзитивна
155 } //
156 else { //якщо матриця ні транзитивна
157     cout << "Нетранзитивна "; //ні антитранзитивна
158 } //то виводимо що матриця нетранзитивна
159 }
160
161 //////////////////////////////////////////////////
162
163
164
165
166
167

```

```

Введіть потужність 1 множини: 3
Введіть потужність 2 множини: 3
Введіть 1 множину: 5 6 7
Введіть 2 множину: 1 2 3
1      1      0
1      1      0
1      1      1
Рефлексивна Несиметрична Транзитивна

```

Відповіді:

Висновок: на цій лабораторній роботі я набув практичних вмінь та навичок при побудові матриць бінарних відношень та визначені їх типів.