

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра систем штучного інтелекту

Лабораторна робота №3

З дисципліни

«Дискретна математика»

Виконав :

Студент КН-113

Сайкевич В.А.

Викладач:

Мельникова Н.І.

Тема : «Побудова матриці бінарного відношення»

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

Варіант №14

Завдання 1

Чи є вірною рівність

$$A \times (B \cap C \cup D) = (A \times B) \cap (A \times C) \cup (A \times D)$$

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

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

$$(x, y, z) \in A \& (x \in B \& y \in C | z \in D) =$$

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

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

Завдання 2

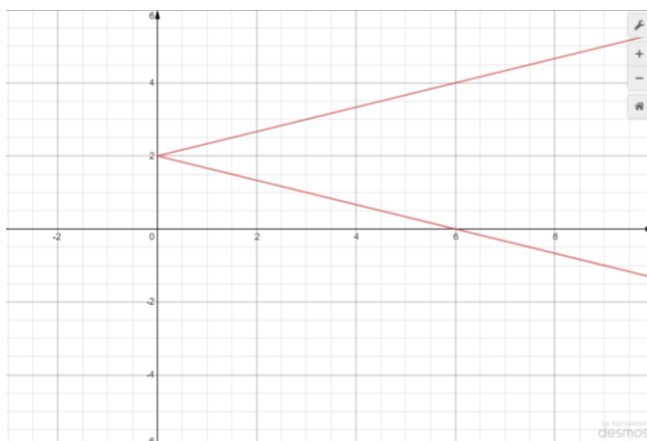
Знайти матрицю відношення $R \subset 2^A \times 2^B$;

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

$B \backslash A$	$\{\emptyset\}$	$\{1\}$	$\{3\}$	$\{1, 3\}$
$\{\emptyset\}$	1	0	0	0
$\{2\}$	1	1	0	0
$\{4\}$	1	1	1	0
$\{2, 4\}$	1	1	1	1

Завдання 3

Зобразити відношення графічно: $a = \{(x, y) | (x, y) \in \mathbb{R}^2 \& |6 - 3y| = x\}$



Завдання 4

Маємо бінарне відношення $R \subset A \times A$, де $A = \{a, b, c, d, e\}$, яке задане своєю матрицею:

$$A(R) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

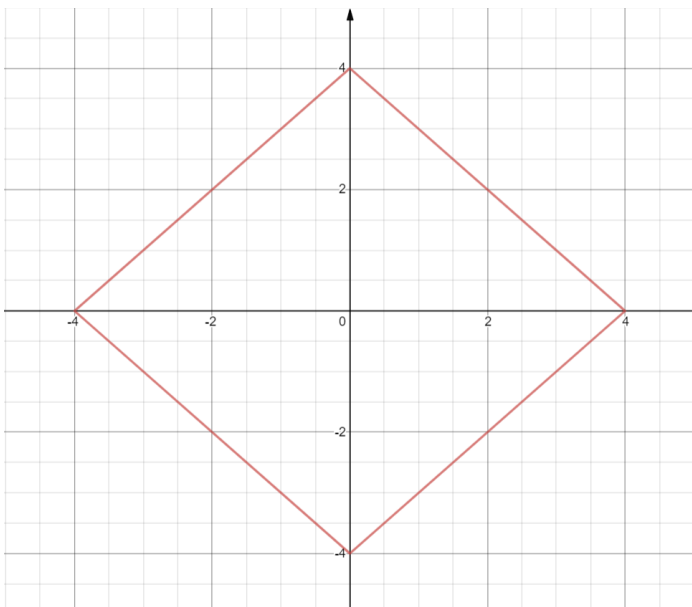
Перевірити чи є дане відношення рефлексивним, симетричним, транзитивним, антисиметричним?

Матриця є рефлексивною (всі елементи на головній діагоналі є 1),
симетричною (кожній 1 є відповідна симетрично розташована 1),
атранзитивною ($a_{22} \wedge a_{24} = a_{24}$, $a_{24} \wedge a_{45} \neq a_{25}$).

Завдання 5

Визначити множину (якщо це можливо), на якій дане відношення є: а) функціональним; б) бієктивним:

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



а) Відношення є функціональним на проміжку $(-4;4)$;

б) Відношення не є бієктивним;

Додаток 2

Завдання : Написати програму, яка знаходить матрицю бінарного відношення $\rho \subset A \times B$, 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.

$$\rho = \{(a, b) \mid a \in A \& b \in B \& a > 2b\};$$

```
1  #include <iostream>
2  #include <math.h>
3  #include <iomanip>
4
5  using namespace std;
6
7  void filling(int size, int* arr) {
8      for (int i = 0; i < size; i++) {
9          cout << i + 1 << " ";
10         cin >> *(arr + i);
11     }
12 }
13
14 bool output(int sizeA, int sizeB, bool** arr, int* arrA, int* arrB) {
15     bool one = 0, null = 0;
16     for (int i = 0; i < sizeB; i++) {
17         for (int j = 0; j < sizeA; j++) {
18             if (arrA[j] > 2 * arrB[i]) { *(arr + i) + j = 1; one = 1; }
19             else if (arrA[j] <= 2 * arrB[i]) { *(arr + i) + j = 0; null = 1; }
20         }
21         cout << "B\\A";
22         for (int i = 0; i < sizeA; i++) {
23             cout << setw(2) << i + 1;
24         }
25         cout << "\n ";
26         for (int i = 0; i < sizeA; i++) {
27             cout << "...";
28         }
29         cout << endl;
30         for (int i = 0; i < sizeB; i++) {
31             cout << setw(2) << i + 1 << " ";
32             for (int j = 0; j < sizeA; j++) {
33                 cout << setw(2) << arr[i][j];
34             }
35             cout << endl;
36         }
37         if (one && null) { return 1; }
38         else if (null) { return 0; }
39     }
40     int reflexive(int size, bool** arr) {
41         bool one = 0, null = 0;
42         for (int i = 0; i < size; i++) {
43             if (arr[i][i]) { one = 1; }
44             else { null = 1; }
45         }
46         if (one && null) { return 0; }
47         else if (one) { return 1; }
48         else if (null) { return -1; }
49     }
50     int symmetry(int size, bool** arr) {
51         bool one = 0, null = 0;
52         for (int i = 0; i < size; i++) {
53             for (int j = 0; j < size; j++) {
54                 if (arr[i][j] == 1 && arr[j][i] == 1 && i != j) { one = 1; }
55                 else if (arr[i][j] != arr[j][i]) { null = 1; }
56             }
57         }
58         if (one && null) { return 0; }
59         else if (one) { return 1; }
60         else if (null) { return -1; }
61     }
62     else { return 1; }
63     int transitive(int sizeA, int sizeB, bool** arr) {
64         bool one = 0, null = 0;
65         for (int i = 0; i < sizeB; i++) {
66             for (int j = 0; j < sizeB && j < sizeA; j++) {
67                 for (int k = 0; k < sizeA; k++) {
68                     if (arr[i][j] == 1 && arr[j][k] == 1 && arr[i][k] == 1) { one = 1; }
69                     else if (arr[i][j] == 1 && arr[j][k] == 1 && arr[i][k] == 0) { null = 1; }
70                 }
71             }
72         }
73         if (one && null) { return 0; }
74         else if (one) { return 1; }
75         else if (null) { return -1; }
76         else { return 1; }
77     }
78
79     void main()
80     {
81         int sizeA, sizeB;
82         cout << "Enter size of array A ";
83         cin >> sizeA;
84         cout << "Enter size of array B ";
85         cin >> sizeB;
86         bool** arr = new bool* [sizeB];
87         int* arrA = new int[sizeA];
88         int* arrB = new int[sizeB];
89         for (int i = 0; i < sizeB; i++) {
90             arr[i] = new bool[sizeA];
```

```

91     }
92     cout << "Enter elements of array A\n";
93     filling(sizeA, arrA);
94     cout << "Enter elements of array B\n";
95     filling(sizeB, arrB);
96     bool null-output(sizeA, sizeB, arr, arrA, arrB);
97     cout << "Matrix is ";
98     if (sizeA == sizeB) {
99         switch (reflexive(sizeA, arr)) {
100             case -1: cout << "antireflexive, "; break;
101             case 0: cout << "areflexive, "; break;
102             case 1: cout << "reflexive, "; break;
103         }
104         if (null) {
105             switch (symmetry(sizeA, arr)) {
106                 case -1: cout << "antisymmetric, "; break;
107                 case 0: cout << "asymmetric, "; break;
108                 case 1: cout << "symmetric, "; break;
109             }
110         }
111     }
112     if (null) {
113         switch (transitive(sizeA, sizeB, arr)) {
114             case -1: cout << "antitransitive\n"; break;
115             case 0: cout << "atransitive\n"; break;
116             case 1: cout << "transitive\n"; break;
117         }
118     }
119 }

```

Результати :

<pre> Enter size of array A 4 Enter size of array B 5 Enter elements of array A 1 2 2 3 3 5 4 6 Enter elements of array B 1 1 2 2 3 3 4 0 5 4 0\A 1 2 3 4 ----- 1 0 1 1 1 2 0 0 1 1 3 0 0 0 0 4 1 1 1 1 5 0 0 0 0 Matrix is atransitive </pre>	<pre> Enter size of array A 4 Enter size of array B 4 Enter elements of array A 1 1 2 2 3 8 4 9 Enter elements of array B 1 0 2 3 3 2 4 6 0\A 1 2 3 4 ----- 1 1 1 1 1 2 0 0 1 1 3 0 0 1 1 4 0 0 0 0 Matrix is areflexive, antisymmetric, transitive </pre>
---	--

Висновок:

Ми набули практичних вмінь та навичок при побудові матриць бінарних відношень та визначені їх типів.