

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

Кафедра систем штучного інтелекту

Лабораторна робота №1
з дисципліни
«Алгоритмізація та програмування»

Виконав:
студент групи КН-115
Гончаренко Н.
Викладач:
Мочурад Л. І.

Львів – 2019 р.

Лабораторна робота № 3.

Тема: Побудова матриці бінарного відношення

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

Варіант 6

1. Чи є вірною рівність: $(A \times B) \cap (C \times D) = (A \cap C) \times (B \cap D)$?

Нехай $(x, y) \in (A \times B) \cap (C \times D) \equiv ((x, y) \in (A \times B)) \cap ((x, y) \in (C \times D))$
 $\equiv (x \in A \& y \in B) \& (x \in C \& y \in D) \equiv (x \in A \cap C) \times (y \in B \cap D) \equiv$
 $(A \cap C) \times (B \cap D)$

2. Знайти матрицю відношення $R \subset 2^B \times A$, де

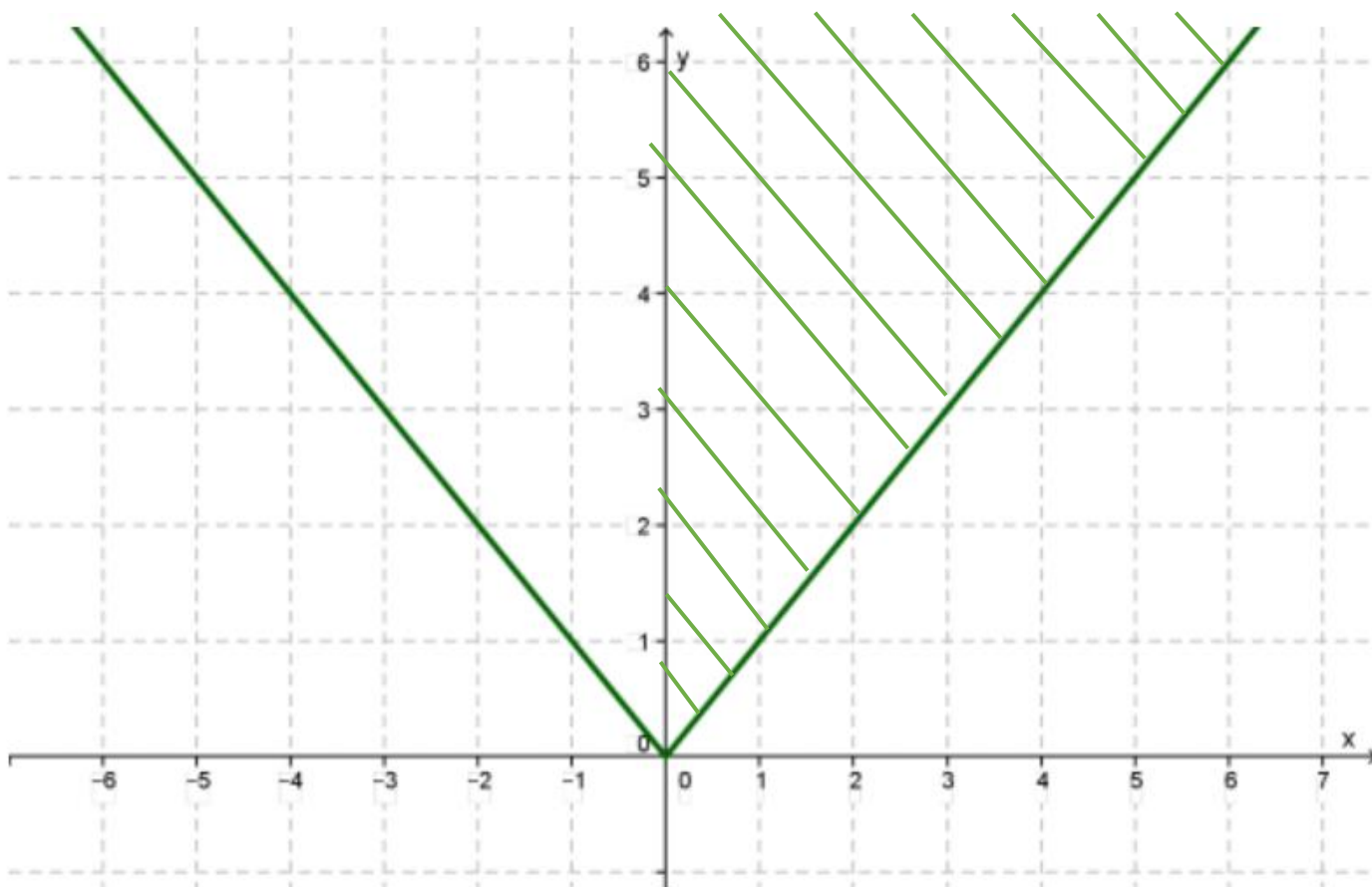
$R = \{(x, y): x \subset B \& y \in A \& |x| = \frac{y}{2}\}$, де $B = \{1, 2\}, A = \{y | y \in \mathbb{Z} \& 1 \leq y \leq 4\}$

\mathbb{Z} – множина цілих чисел.

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

3. Зобразити відношення графічно:

$a = \{(x, y) | (x, y) \in R^2 \& x \leq y\}$, де R – множина дійсних чисел.



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

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

Рефлексивним : Так.

Симетричним: Так.

Транзитивним: Так.

Антисиметричним: Ні.

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

б) бієктивним: $a = \{(x, y) \mid (x, y) \in R^2 \text{ \& } (x + y)^3 = 5\}.$

Множина:

$$A = \{0, \frac{\sqrt[3]{5}}{3}, \frac{\sqrt[3]{5}}{4}, \dots, \frac{\sqrt[3]{5}}{2}, \dots, \frac{2\sqrt[3]{5}}{3}, \frac{3\sqrt[3]{5}}{4}, \sqrt[3]{5}\}$$

Завдання №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.

$$6. \quad \rho = \{(a, b) \mid a \in A \ \& \ b \in B \ \& \ 2a < b\};$$

```
1  #include <iostream>
2  #include <fstream>
3  #include "windows.h"
4
5
6
7
8  using namespace std;
9
10 int Name_Array = 65;
11
12
13 double* File_List(ifstream* read, int); //-----Створення Множини-----
14 void Matrix(double a[], double b[], int power); //-----Створення і відображення матриці-----
15 double* Create_List(int); //-----Створення Множини-----
16 void Refl(double** Matrix, int N); //-----Рефлексивність-----
17 void Simetric(double** Matrix, int N); //-----Симетричність-----
18 void Tranz(double** Matrix, int N); //-----Транзитивність-----
19 void File_type(); // -----Введення файлами -----
20 void Hard_Type(); // -----Введення в ручну -----
21
22
23
24 int main()
25 {
26     cout << "Enter number for choose:"
27         << "\nFor write type enter    1"
28         << "\nFor file type enter      2\nEnter: ";
29     int k = 11;
30     cin >> k;
31     if ( k == 1 ) Hard_Type();
32     else if (k == 2) File_type();
33     else
34     {
```

```

35         cerr << "Error";
36         exit(2);
37     }
38     return 0;
39 }
40
41
42 //template <class A, class P>
43 void Matrix(double a[], double b[], int power)
44 {
45     double** Matrix = new double*[power];
46     for (int i = 0; i < power; i++)
47     {
48         Matrix[i] = new double[power];
49     }
50     for (int y = 0; y < power; y++)
51     {
52         for (int x = 0; x < power; x++)
53         {
54             if (2 * a[x] < b[y]) Matrix[x][y] = 1;
55             else Matrix[x][y] = 0;
56         }
57     }
58
59
60     for (int y = 0; y < power; y++)
61     {
62         for (int x = 0; x < power; x++)
63         {
64             cout << Matrix[x][y] << " ";
65         }
66         cout << "\n" << endl;
67     }
68
69     Refl(Matrix, power);
70     Simetric(Matrix, power);
71     Tranz(Matrix, power);
72     cout << "\n\n";
73     delete[] Matrix;
74 }
75
76
77
78 double* Create_List(int n)
79 {
80     cout << "Enter " << n << " elements for array " << char(Name_Array) << " : ";
81     ::Name_Array++;
82     double* List = new double[n];
83     for (int i = 0; i < n; i++)
84     {
85         cin >> List[i];
86     }
87     return List;
88 }
89
90
91
92
93
94 void Refl(double** Matrix, int N)
95 {
96     SetConsoleCP(1251);
97     SetConsoleOutputCP(1251);
98     int temp = 0;
99     for (int i = 0; i < N; i++)
100     {

```

```

101     if (Matrix[i][i] == 0)
102     {
103         cout << "Не рефлексивне.\n";
104         temp = 1;
105         break;
106     }
107 }
108 if (temp == 0) cout << "Рефлексивне.\n";
109 }
110
111 void Simetric(double** Matrix, int N)
112 {
113     int temp = 0, T=0;
114     SetConsoleCP(1251);
115     SetConsoleOutputCP(1251);
116     // -----Симетрична----- //
117     for (int i = 0; i < N; i++)
118     {
119         for (int j = 0; j < N; j++)
120         {
121             if (Matrix[i][j] == 1 && i != j) T++;
122             if ((Matrix[i][j] == 1) && (Matrix[j][i] != 1))
123             {
124                 temp = 1;
125                 goto antisim;
126                 break;
127             }
128         }
129     }
130     goto end;
131     // -----Антисиметрична----- //
132     antisim:
133     for (int i = 0; i < N; i++)
134     {
135         for (int j = 0; j < N; j++)
136         {
137             if (Matrix[i][j] == 1 && Matrix[j][i] == 1 && i != j)
138             {
139                 temp = 2;
140                 break;
141             }
142         }
143     }
144     end:
145     if (temp == 0 && T != 0) cout << "Симетричне.\n";
146     if (temp == 1) cout << "Антисиметричне.\n";
147     if (temp == 2 || (temp == 0 && T == 0)) cout << "Асиметричне.\n";
148 }
149
150
151 void Tranz(double** Matrix, int N)
152 {
153     int temp = 0, T=0;
154     SetConsoleCP(1251);
155     SetConsoleOutputCP(1251);
156
157     for (int i = 0; i < N; i++)
158     {
159         for (int j = 0; j < N; j++)
160         {
161             for (int k = 0; k < N; k++)
162             {
163                 if ((Matrix[i][j] == 1) && (Matrix[j][k] == 1))

```

```

163         if ((Matrix[i][j] == 1) && ( Matrix[j][k] == 1))
164         {
165             if (Matrix[i][k] != 1)
166             {
167                 temp = 1;
168                 break;
169             }
170         }
171     }
172 }
173 }
174 if (temp == 0) cout << "Транзитивне.\n";
175 if (temp == 1) cout << "Не транзитивне.\n";
176 }
177
178
179 double* File_List(ifstream* read,int power)
180 {
181     double* List_f = new double[power];
182     for (int i=0; i < power; i++)
183     {
184         *read >> List_f[i];
185     }
186     return List_f;
187 }
188
189
190 void File_type()
191 {
192     int x;
193     ifstream read1("test1.txt"); //Не рефлексивне. Асиметричне. Транзитивне.
194     read1 >> x;
195     Matrix(File_List(&read1, x), File_List(&read1, x), x);
196     ifstream read2("test2.txt"); //Рефлексивне Асиметричне Транзитивне
197     read2 >> x;
198     Matrix(File_List(&read2, x), File_List(&read2, x), x);
199     ifstream read3("test3.txt"); //Не рефлексивне Антисиметричне Транзитивне
200     read3 >> x;
201     Matrix(File_List(&read3, x), File_List(&read3, x), x);
202     ifstream read4("test4.txt"); //Не рефлексивне. Асиметричне. Не транзитивне.
203     read4 >> x;
204     Matrix(File_List(&read4, x), File_List(&read4, x), x);
205 }
206 void Hard_Type()
207 {
208     int x;
209     cout << "Enter a power of arraies : ";
210     cin >> x;
211     Matrix(Create_List(x), Create_List(x), x);
212 }

```