МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский государственный университет им. А.Н. Косыгина   
(Технологии. Дизайн. Искусство)»

Кафедра автоматизированных систем обработки информации и управления

Отчет по лабораторной работе №2

# по дисциплине «Теория множеств и алгоритмы на графах».

Тема: «Бинарные отношения»

Выполнил: Юрков Д.А., МВА-122

Проверил: Кузьмина Т. М.

Москва 2024

**Задание**

1. Написать программу, которая демонстрирует бинарное отношение на множестве как некоторое подмножество декартово квадрата этого множества (А×А). Для этого задать некоторое множество чисел. Это множество можно задать один раз внутри программного кода, но при защите лабораторной работы, преподаватель может попросить изменить его. На экран вывести элементы декартова квадрата исходного множества, причем элементы, составляющие заданное вариантом отношение должны быть каким-то образом выделены (например, цветом).

2. Определить свойства исследуемого отношения.

Множество А – конечное множество целых чисел, a, b ϵ A.

**Индивидуальное задание**

Элемент a находится в отношении Р к элементу b, если a + 2 < b;

**Текст программы**

using System;

using System.Collections.Generic;

using System.Collections.ObjectModel;

using System.Globalization;

using System.Linq;

using System.Windows;

using System.Windows.Data;

using System.Windows.Media;

namespace Algos2Lab

{

public partial class MainWindow : Window

{

private readonly List<int> \_setA = new List<int> { 0, 1, 3, 4, 5 };

public MainWindow()

{

InitializeComponent();

DisplayRelation();

DisplayProperties();

}

private void DisplayRelation()

{

var data = new ObservableCollection<RelationRow>();

foreach (var row in \_setA.Select(b => new RelationRow

{

B = b,

A0 = (0 + 2 < b) ? $"({0}, {b})" : "",

A1 = (1 + 2 < b) ? $"({1}, {b})" : "",

A3 = (3 + 2 < b) ? $"({3}, {b})" : "",

A4 = (4 + 2 < b) ? $"({4}, {b})" : "",

A5 = (5 + 2 < b) ? $"({5}, {b})" : ""

}))

{

data.Add(row);

}

RelationDataGrid.ItemsSource = data;

}

private void DisplayProperties()

{

var reflexive = true;

var antiReflexive = true;

var symmetric = true;

var antiSymmetric = true;

var transitive = true;

foreach (var t in \_setA)

{

foreach (var t1 in \_setA)

{

var relation = t + 2 < t1;

if (t == t1)

{

if (relation)

{

reflexive = false;

}

}

else

{

if (relation)

{

antiReflexive = false;

}

if (relation && !(t1 + 2 < t))

{

symmetric = false;

}

if (relation && (t1 + 2 < t) && (t != t1))

{

antiSymmetric = false;

}

}

}

}

foreach (var relation1 in from t in \_setA from t1 in \_setA from t2 in \_setA let relation1 = t + 2 < t1 let relation2 = t1 + 2 < t2 let relation3 = t + 2 < t2 where relation1 && relation2 && !relation3 select relation1)

{

transitive = false;

}

label1.Content = $"Рефлексивное: {reflexive}\n" +

$"Антирефлексивное: {antiReflexive}\n" +

$"Симметричное: {symmetric}\n" +

$"Антисимметричное: {antiSymmetric}\n" +

$"Транзитивное: {transitive}";

}

public class RelationRow

{

public int B { get; set; }

public string A0 { get; set; }

public string A1 { get; set; }

public string A3 { get; set; }

public string A4 { get; set; }

public string A5 { get; set; }

}

}

public class BackgroundConverter : IValueConverter

{

public object Convert(object value, Type targetType, object parameter, CultureInfo culture)

{

if (value is string content && !string.IsNullOrEmpty(content))

{

return Brushes.LightGreen;

}

return Brushes.White;

}

public object ConvertBack(object value, Type targetType, object parameter, CultureInfo culture)

{

throw new **NotImplementedException**();

}

}

}

Скрин демонстрирующий работу программы представлен на рис. 1.

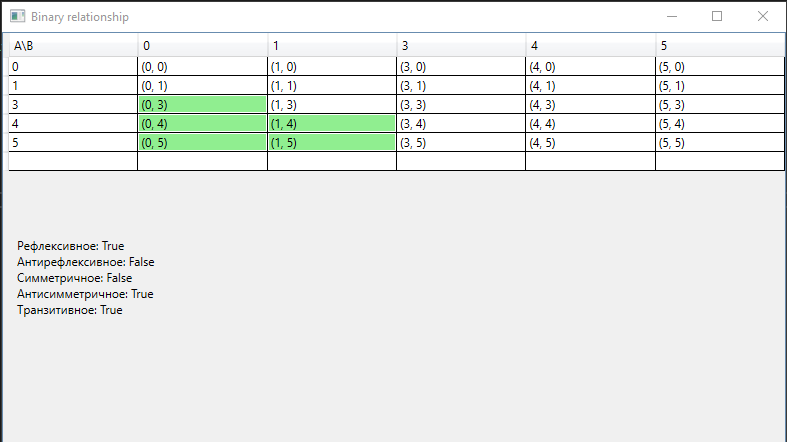


Рис. 1. Скрин работы программы.

**Описание свойств, исследуемого отношения**

Множество A: {0, 1, 3, 4, 5}

Отношение R: {(0,3), (0,4), (0,5), (1,4), (1,5)}

Свойства:

Рефлексивность: Отношение не является рефлексивным, так как нет ни одного элемента (a, a) в R.

Антирефлексивность: Отношение является антирефлексивным, так как нет ни одного элемента (a, a) в R.

Симметричность: Отношение не является симметричным, так как, например, (0,3) ∈ R, но (3,0) ∉ R.

Антисимметричность: Отношение является антисимметричным, так как нет пар (a, b) и (b, a) в R с a ≠ b.

Транзитивность: Отношение является транзитивным, так как для всех (a, b) и (b, c) в R, (a, c) также в R.

**Выводы**

Программа успешно идентифицирует и подсвечивает ячейки, удовлетворяющие условию a + 2 < b.

Отношение, определенное на множестве A с условием a + 2 < b, является антирефлексивным, антисимметричным и транзитивным, но не является рефлексивным или симметричным.

Результаты подтверждают корректность реализации и анализа бинарного отношения в программе.