Кам’янець-Подільський національний університет  
імені Івана Огієнка

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

ЛАБОРАТОРНА РОБОТА #0110

з дисципліни «Об’єктно-орієнтоване програмування»

**Перевантаження операцій**

**Виконала**:  
студентка 2-го курса  
гр. KN1-B23  
Серебрянський Артем

**Прийняв**:  
старший викладач,  
Слободянюк О.В.

Дата здачі 09.05.2025р.

Кам’янець-Подільський – 2025

**Варіант: 3**Абстрактний тип даних (АТД ) – множина з елементами типу **char.**Додатково перенавантажувати наступні операції:

“-” - видалити елемент з множини(типу set-char);

“>”- перевірка на підмножину;

“!=” - перевірка множин на нерівність.

**Покликання на Git Hub:**

**Код програми коли уводиться все з клавіатури:**using System;

using System.Collections.Generic;

using System.Text;

public class МножинаСимволів

{

private HashSet<char> елементи;

public МножинаСимволів() => елементи = new HashSet<char>();

public МножинаСимволів(params char[] символи) => елементи = new HashSet<char>(символи);

public void Додати(char символ) => елементи.Add(символ);

public void Видалити(char символ) => елементи.Remove(символ);

public static МножинаСимволів operator -(МножинаСимволів множина, char символ)

{

var результат = new МножинаСимволів();

foreach (var елемент in множина.елементи)

if (елемент != символ) результат.Додати(елемент);

return результат;

}

public static bool operator >(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.IsSupersetOf(множина2.елементи);

public static bool operator <(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.IsSubsetOf(множина2.елементи);

public static bool operator !=(МножинаСимволів множина1, МножинаСимволів множина2)

=> !множина1.елементи.SetEquals(множина2.елементи);

public static bool operator ==(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.SetEquals(множина2.елементи);

public override bool Equals(object obj) => obj is МножинаСимволів інша && this == інша;

public override int GetHashCode() => елементи.GetHashCode();

public override string ToString() => "{" + string.Join(", ", елементи) + "}";

}

class Програма

{

static void Main()

{

Console.OutputEncoding = Encoding.UTF8;

Console.InputEncoding = Encoding.UTF8;

Console.WriteLine("Введіть елементи для множини A (через пробіл):");

var введенняA = Console.ReadLine();

var множинаA = new МножинаСимволів(введенняA.ToCharArray());

Console.WriteLine("Введіть елементи для множини B (через пробіл):");

var введенняB = Console.ReadLine();

var множинаB = new МножинаСимволів(введенняB.ToCharArray());

Console.WriteLine("Введіть елементи для множини C (через пробіл):");

var введенняC = Console.ReadLine();

var множинаC = new МножинаСимволів(введенняC.ToCharArray());

Console.WriteLine($"Множина A: {множинаA}");

Console.WriteLine($"Множина B: {множинаB}");

Console.WriteLine($"Множина C: {множинаC}");

Console.WriteLine("Введіть символ для видалення з множини A:");

char символДляВидалення = Console.ReadLine()[0];

var множинаD = множинаA - символДляВидалення;

Console.WriteLine($"Множина A без '{символДляВидалення}': {множинаD}");

Console.WriteLine($"Множина A > Множина B: {множинаA > множинаB}");

Console.WriteLine($"Множина B > Множина A: {множинаB > множинаA}");

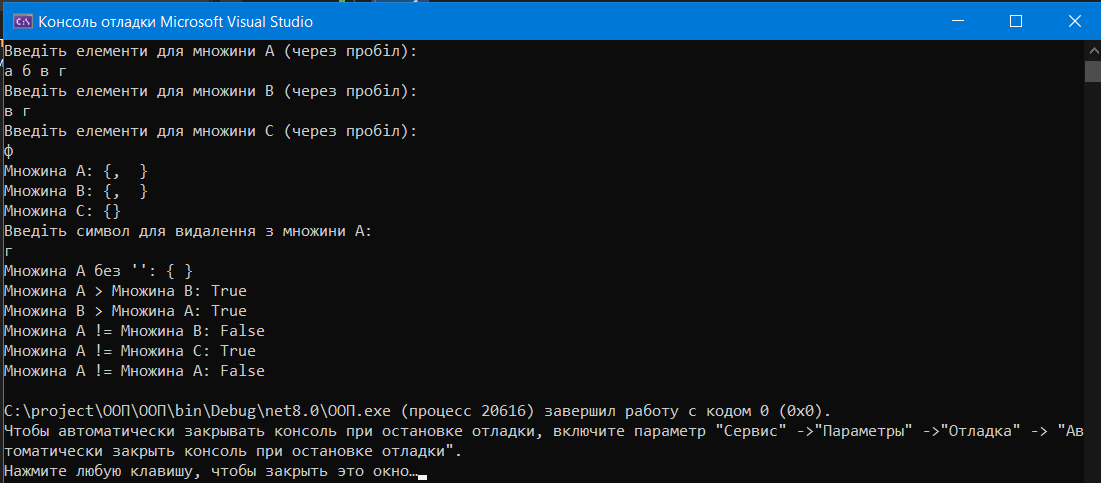
Console.WriteLine($"Множина A != Множина B: {множинаA != множинаB}");

Console.WriteLine($"Множина A != Множина C: {множинаA != множинаC}");

Console.WriteLine($"Множина A != Множина A: {множинаA != множинаA}");

}

}

**Результат:  
**

**Код програми коли все заддано:**using System;

using System.Collections.Generic;

using System.Text;

public class МножинаСимволів

{

private HashSet<char> елементи;

public МножинаСимволів() => елементи = new HashSet<char>();

public МножинаСимволів(params char[] символи) => елементи = new HashSet<char>(символи);

public void Додати(char символ) => елементи.Add(символ);

public void Видалити(char символ) => елементи.Remove(символ);

public static МножинаСимволів operator -(МножинаСимволів множина, char символ)

{

var результат = new МножинаСимволів();

foreach (var елемент in множина.елементи)

if (елемент != символ) результат.Додати(елемент);

return результат;

}

public static bool operator >(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.IsSupersetOf(множина2.елементи);

public static bool operator <(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.IsSubsetOf(множина2.елементи);

public static bool operator !=(МножинаСимволів множина1, МножинаСимволів множина2)

=> !множина1.елементи.SetEquals(множина2.елементи);

public static bool operator ==(МножинаСимволів множина1, МножинаСимволів множина2)

=> множина1.елементи.SetEquals(множина2.елементи);

public override bool Equals(object obj) => obj is МножинаСимволів інша && this == інша;

public override int GetHashCode() => елементи.GetHashCode();

public override string ToString() => "{" + string.Join(", ", елементи) + "}";

}

class Програма

{

static void Main()

{

Console.OutputEncoding = Encoding.UTF8;

Console.InputEncoding = Encoding.UTF8;

var множинаA = new МножинаСимволів('а', 'б', 'в', 'г');

var множинаB = new МножинаСимволів('б', 'в');

var множинаC = new МножинаСимволів('а', 'б', 'в');

Console.WriteLine($"Множина A: {множинаA}");

Console.WriteLine($"Множина B: {множинаB}");

Console.WriteLine($"Множина C: {множинаC}");

var множинаD = множинаA - 'а';

Console.WriteLine($"Множина A - 'а': {множинаD}");

Console.WriteLine($"Множина A > Множина B: {множинаA > множинаB}");

Console.WriteLine($"Множина B > Множина A: {множинаB > множинаA}");

Console.WriteLine($"Множина A != Множина B: {множинаA != множинаB}");

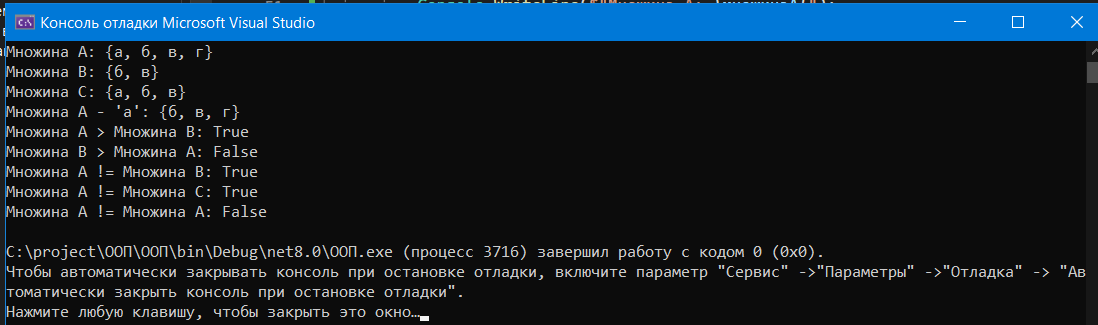
Console.WriteLine($"Множина A != Множина C: {множинаA != множинаC}");

Console.WriteLine($"Множина A != Множина A: {множинаA != множинаA}");

}

}

**Результат:**

****