**2 Классы**

Задание 1. Создать класс А с целочисленными полями а и b и двумя методами согласно варианту. Внутри класса реализовать конструктор для инициализации a и b. Создать объект класса и продемонстрировать работу со всеми элементами класса. Метод вычисления значения выражения , метод возведения в куб суммы a и b.

Листинг программы:

class A {

int a; int b;

public A(int a, int b){

this.a = a; this.b = b;}

public double ExpressionValue() {

double result = (-b + (1.0 / a)) / 3.0; return result; }

public int SumCubed() {

int sum = a + b; int cubedSum = sum \* sum \* sum; return cubedSum;}

}

class a1 {

static void Main(string[] args) {

A obj = new A(2, 3);

double exprValue = obj.ExpressionValue();

Console.WriteLine($"Значение выражения ((-b+(1/a))/3) равно {exprValue}.");

int cubedSum = obj.SumCubed();

Console.WriteLine($"Куб суммы a и b равен {cubedSum}."); }}

Таблица 2.1 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Значение выражения равно -0,8333333333333334.  Куб суммы a и b равен 125. |

Источник: собственная разработка

Анализ результатов:

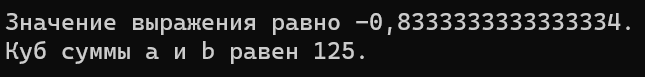


Рисунок 2.1 – Результат работы программы

Источник: собственная разработка

Задание 2. Построить иерархию классов в соответствии с вариантом задания: Журнал, книга, печатное издание, учебник.

Листинг программы:

class PrintPublication {

public string Title { get; set; }

public string Author { get; set; }

public int Year { get; set; }

public PrintPublication(string title, string author, int year) {

Title = title; Author = author; Year = year; }

public virtual void PrintInfo() {

Console.WriteLine($"Название: {Title}, Автор: {Author}, Год: {Year}"); }}

class Book : PrintPublication {

public string Publisher { get; set; }

public Book(string title, string author, int year, string publisher) : base(title, author, year) {

Publisher = publisher; }

public override void PrintInfo() {

Console.WriteLine($"Книга: {Title}, Автор: {Author}, Год: {Year}, Издательство: {Publisher}");}}

class Magazine : PrintPublication {

public int Number { get; set; }

public Magazine(string title, string author, int year, int number) : base(title, author, year) {

Number = number; }

public override void PrintInfo(){

Console.WriteLine($"Журнал: {Title}, Автор: {Author}, Год: {Year}, Номер: {Number}"); } }

class Textbook : Book {

public string Subject { get; set; }

public Textbook(string title, string author, int year, string publisher, string subject) : base(title, author, year, publisher){

Subject = subject;}

public override void PrintInfo() {

Console.WriteLine($"Учебник: {Title}, Автор: {Author}, Год: {Year}, Издательство: {Publisher}, Предмет: {Subject}"); } }

class a2{

static void Main(string[] args) {

PrintPublication publication = new PrintPublication("Моя книга", "Я", 2022);

publication.PrintInfo();

Book book = new Book("Фотосессия без премий", "Варченя Артём", 2022, "ГГКТТИД");

book.PrintInfo();

Magazine magazine = new Magazine("Всё о Геншине", "Беленица Людмила", 2022, 1);

magazine.PrintInfo();

Textbook textbook = new Textbook("Тупой и умный c#", "Саливончик Антон", 2022, "Сам сделал", "Информатика");

textbook.PrintInfo(); }}

Таблица 2.2 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Название: Моя книга, Автор: Я, Год: 2022  Книга: Фотосессия без премий, Автор: Варченя Артём, Год: 2022, Издательство: ГГКТТИД  Журнал: Всё о Геншине, Автор: Беленица Людмила, Год: 2022, Номер: 1  Учебник: Тупой и умный c#, Автор: Саливончик Антон, Год: 2022, Издательство: Сам сделал, Предмет: Информатика |

Источник: собственная разработка

Анализ результатов:

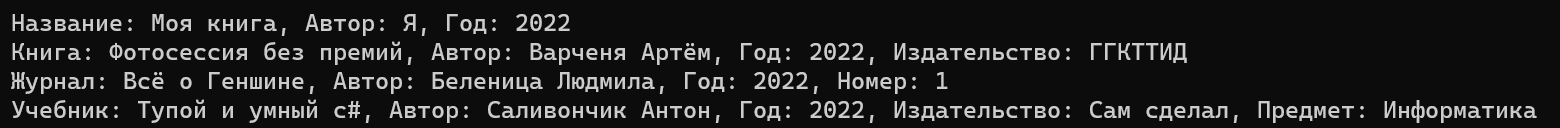


Рисунок 2.2 – Результат работы программы

Источник: собственная разработка

Задание 3. Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменение размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольниками прямоугольника, являющегося общей частью (пересечением) двух прямоугольников. Написать программу, демонстрирующую все разработанные элементы класса.

Листинг программы:

class Rectangle{

public Rectangle(int x, int y, int width, int height) {

X = x; Y = y; Width = width; Height = height;}

public int X{get; set;}

public int Y{get;set;}

public int Width{get;set;}

public int Height {get;set;}

public void Offset(int x, int y) {X += x; Y += y;}

public static Rectangle Intersect(Rectangle a, Rectangle b) {

int maxStartX = Math.Max(a.X, b.X);

int minEndX = Math.Min(a.X + a.Width, b.X + b.Width);

int MaxStartY = Math.Max(a.Y, b.Y);

int minEndY = Math.Min(a.Y + a.Height, b.Y + b.Height);

if (minEndX >= maxStartX && minEndY >= MaxStartY) {

return new Rectangle(maxStartX, MaxStartY, minEndX - maxStartX, minEndY - MaxStartY); }

return null;}

public static Rectangle Union(Rectangle a, Rectangle b) {

int minStartX = Math.Min(a.X, b.X);

int maxEndX = Math.Max(a.X + a.Width, b.X + b.Width);

int minStartY = Math.Min(a.Y, b.Y);

int MaxEndY = Math.Max(a.Y + a.Height, b.Y + b.Height);

return new Rectangle(minStartX, minStartY, maxEndX - minStartX, MaxEndY - minStartY); }

public override string ToString() {

return String.Format("X:{0} Y:{1} W:{2} H:{3}", X, Y, Width, Height);}}

class Program {

static void Main(string[] args) {

Rectangle r1 = new Rectangle(0, 0, 10, 10);

Console.WriteLine("Исходный прямоугольник: " + r1);

Console.Write("Смещённый прямоугольник: ");

r1.Offset(5, 0); Console.WriteLine(r1);

Rectangle r2 = new Rectangle(2, 2, 20, 20);

Console.WriteLine("Пересечение прямоугольников: " + Rectangle.Intersect(r1, r2));

Console.WriteLine("Объединение прямоугольников: " + Rectangle.Union(r1, r2));}}

Таблица 2.3 – Выходные и входные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Исходный прямоугольник: X:0 Y:0 W:10 H:10  Смещённый прямоугольник: X:5 Y:0 W:10 H:10  Пересечение прямоугольников: X:5 Y:2 W:10 H:8  Объединение прямоугольников: X:2 Y:0 W:20 H:22 |

Источник: собственная разработка

Анализ результатов:

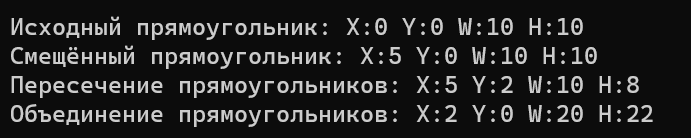


Рисунок 2.3 – Результат работы программы

Источник: собственная разработка