Практическая работа №13

Вариант 1

Выполнял: Шеменев | Нестеренко, ИСП-3-27

1.

using System;

class Shemenevnesterenko

{

static void Main()

{

// Ввод количества уроков

Console.Write("Введите количество уроков: ");

int lessons = int.Parse(Console.ReadLine());

// Параметры времени

int lessonDuration = 45;

int breakDuration = 10;

// Расчет общего времени

// Уроки: lessons \* 45 минут

// Перемены: (lessons - 1) \* 10 минут (перемен на 1 меньше чем уроков)

int totalMinutes = lessons \* lessonDuration + (lessons - 1) \* breakDuration;

// Перевод в часы и минуты

int hours = totalMinutes / 60;

int minutes = totalMinutes % 60;

// Вывод результата

Console.WriteLine($"Общее время в техникуме: {hours} ч {minutes} мин");

Console.WriteLine($"Всего минут: {totalMinutes}");

}

}

2.

using System;

class Shemenevnesterenko

{

static void Main()

{

// Ввод значения a

Console.Write("Введите значение a: ");

double a = double.Parse(Console.ReadLine());

// Вычисление выражения sqrt(a + 2)

double result = Math.Sqrt(a + 2);

// Вывод результата

Console.WriteLine($"√({a} + 2) = {result:F2}");

// Проверка на отрицательное подкоренное выражение

if (a + 2 < 0)

{

Console.WriteLine("Ошибка: нельзя извлечь корень из отрицательного числа!");

}

}

}

3.

using System;

class Shemenevnesterenko

{

static void Main()

{

// Вывод заголовка программы

Console.WriteLine("Вычисление периметра и площади треугольника.");

// Ввод исходных данных

Console.WriteLine("Введите исходные данные:");

Console.Write("a = ");

double a = double.Parse(Console.ReadLine());

Console.Write("b = ");

double b = double.Parse(Console.ReadLine());

Console.Write("c = ");

double c = double.Parse(Console.ReadLine());

// Проверка на существование треугольника

if (a + b > c && a + c > b && b + c > a)

{

// Вычисление полупериметра

double p = (a + b + c) / 2;

// Вычисление площади по формуле Герона

double S = Math.Sqrt(p \* (p - a) \* (p - b) \* (p - c));

// Вывод результатов в соответствии с требуемым форматом

Console.WriteLine("\nРезультат:");

Console.WriteLine($"P = ({a}+{b}+{c})/2={p}");

Console.WriteLine($"S = {S:F6}");

}

else

{

Console.WriteLine("\nОшибка: треугольник с такими сторонами не существует!");

Console.WriteLine("Сумма любых двух сторон должна быть больше третьей стороны.");

}

}

}

4.

using System;

class Shemenevnesterenko

{

static void Main()

{

Console.WriteLine("Вычислить продолжительность года на двух планетах");

// Планета №1

Console.WriteLine("\nВведите исходные данные для планеты №1:");

Console.Write("Введите название планеты №1: ");

string planet1 = Console.ReadLine();

Console.Write("Радиус орбиты (млн. км): r1 = ");

double r1\_million = double.Parse(Console.ReadLine());

Console.Write("Орбитальная скорость (км/с): v1 = ");

double v1 = double.Parse(Console.ReadLine());

// Перевод млн. км в км

double r1\_km = r1\_million \* 1000000;

Console.WriteLine($"Переводом млн. км в км: r1 = r1 \* 1000000 = {r1\_km}");

// Расчет продолжительности года в секундах

double year1\_seconds = 2 \* Math.PI \* r1\_km / v1;

Console.WriteLine($"Продолжительность года: T1 = 2 \* π \* r1 / v1 = {year1\_seconds:F0}");

// Перевод секунд в дни

double year1\_days = year1\_seconds / (60 \* 60 \* 24);

Console.WriteLine($"Переводом секунды в дни: T1 = T1 / (60\*60\*24) = {year1\_days:F6}");

// Планета №2

Console.WriteLine("\nВведите исходные данные для планеты №2:");

Console.Write("Введите название планеты №2: ");

string planet2 = Console.ReadLine();

Console.Write("Радиус орбиты (млн. км): r2 = ");

double r2\_million = double.Parse(Console.ReadLine());

Console.Write("Орбитальная скорость (км/с): v2 = ");

double v2 = double.Parse(Console.ReadLine());

// Перевод млн. км в км

double r2\_km = r2\_million \* 1000000;

Console.WriteLine($"Переводом млн. км в км: r2 = r2 \* 1000000 = {r2\_km}");

// Расчет продолжительности года в секундах

double year2\_seconds = 2 \* Math.PI \* r2\_km / v2;

Console.WriteLine($"Продолжительность года: T2 = 2 \* π \* r2 / v2 = {year2\_seconds:F3}");

// Перевод секунд в дни

double year2\_days = year2\_seconds / (60 \* 60 \* 24);

Console.WriteLine($"Переводом секунды в дни: T2 = T2 / (60\*60\*24) = {year2\_days:F4}");

// Сводная таблица результатов

Console.WriteLine("\n" + new string('=', 60));

Console.WriteLine("СВОДНАЯ ТАБЛИЦА РЕЗУЛЬТАТОВ");

Console.WriteLine(new string('=', 60));

Console.WriteLine($"Планета | Продолжительность года (дней)");

Console.WriteLine(new string('-', 60));

Console.WriteLine($"{planet1,-14} | {year1\_days,28:F2}");

Console.WriteLine($"{planet2,-14} | {year2\_days,28:F2}");

}

}

5.

using System;

class Shemenevnesterenko

{

// Константа ускорения свободного падения

const double g = 9.8; // м/с²

static void Main()

{

Console.WriteLine("Расчет времени падения яблока на голову Ньютона");

// Ввод высоты ветки

Console.Write("Введите высоту ветки (в метрах): ");

double height = double.Parse(Console.ReadLine());

// Проверка корректности высоты

if (height <= 0)

{

Console.WriteLine("Ошибка: высота должна быть положительным числом!");

return;

}

// Расчет времени падения по формуле: t = √(2h/g)

double time = Math.Sqrt(2 \* height / g);

// Вывод результатов

Console.WriteLine("\nРезультаты расчета:");

Console.WriteLine($"Ускорение свободного падения: g = {g} м/с²");

Console.WriteLine($"Высота падения: h = {height} м");

Console.WriteLine($"Время падения: t = √(2\*{height}/{g}) = {time:F2} секунд");

// Дополнительная информация

Console.WriteLine($"\nВремя падения: {time:F4} с");

Console.WriteLine($"Время падения: {time \* 1000:F0} мс");

// Сравнение с реальными значениями

if (time < 1)

{

Console.WriteLine("Яблоко упадет очень быстро!");

}

else if (time < 3)

{

Console.WriteLine("У Ньютона есть время подумать!");

}

else

{

Console.WriteLine("Это очень высокое дерево!");

}

}

}