ООО «КФ ГУМРФ»

Протокол оперативного совещания № 41-ИС

|  |  |
| --- | --- |
| г. Котлас | 26.02.2025 |

О приоритетных целях на неделю 26.02.2025 — 31.06.2025.

Председатель — технический директор ООО «КФ ГУМРФ» Скворцов С. Е.

Рецензист—ООО «КФ ГУМРФ» Таранин А.А.

Рецензист—ООО «КФ ГУМРФ» Шарыпов В.А.

Секретарь — бухгалтер Краев И. А.

Автор — Салыкин Н. В.

Присутствовали:

Таранин А.А

Шарыпов В.А

Краев И.А

Салыкин Н.В

Повестка дня:

1. Описание кода в ответно-вопросной форме

2. Утверждение порядка проведения оперативных совещаний.

Постановили:

1. Заключить договор с организацией ООО «КФ ГУМРФ» на описание кода.

2. Поручить Краеву И. А. записать все описание кода.

Голосование: единогласно за. Проголосовавших против и воздержавшихся нет.

|  |  |  |
| --- | --- | --- |
| Председатель | *Скворцов С.Е* | . |

|  |  |  |
| --- | --- | --- |
| Секретарь | *Краев* | Краев И. А. |

**Вопрос 1**

**Что делают директивы?**

using System.Collections.Generic;

Данная строка портирует класс для работы с такими коллекциями как <List>. Остальные также портируют нужные классы для работы программы.

**Вопрос 2**

**Что делают константы Width и Height?**

const int Width = 40;

const int Height = 20;

Они определяют размер игрового поля, а именно ширину и высоту.

**Вопрос 3**

**За что отвечает класс Point?**

class Point

{

public int X, Y;

public Point(int x, int y) { X = x; Y = y; }

}  
Представляет собой определение публичных переменных X и Y, которые представляют

координаты X и Y на игровом поле. Они нужны для определения и положения змейки и еды.

**Вопрос 4**

**Зачем нужен ListPointBody в классе Snake?**

class Snake

{

public List<Point> Body = new List<Point>();

public char Symbol = '1';

Этот список хранит в виде символа “1” сегмент змейки. Каждый элемент списка – объект Point с координатами X и Y.

**Вопрос 5**

**Зачем нужен метод Move?**

public void Move(int dx, int dy)

{

Point head = new Point(Body[0].X + dx, Body[0].Y + dy);

if (head.X < 0) head.X = Width - 1;

if (head.X >= Width) head.X = 0;

if (head.Y < 0) head.Y = Height - 1;

if (head.Y >= Height) head.Y = 0;

Body.Insert(0, head);

Body.RemoveAt(Body.Count - 1);

}

Метод перемещает змейку.

**Вопрос 6**

**Зачем нужен метод Draw?**

public void Draw()

{

Console.ForegroundColor = ConsoleColor.Green;

foreach (Point p in Body)

{

Console.SetCursorPosition(p.X, p.Y);

Console.Write(Symbol);

}

Console.ResetColor();

}

}

Метод Draw выводит символ змейки на соответствующие координаты консоли, также определяет цвет змейки.

**Вопрос 7**

**Какова роль класса Food?**

class Food

{

public int X, Y;

public char Symbol = '$';

public Food(int x, int y) { X = x; Y = y; }

Класс Food определяет координаты местоположения еды, а также её символьное обозначение в виде знака “$”

**Вопрос 8**

**Зачем необходим второй метод Draw?**

Данный метод также определяет координаты расположения еды, и окрашивает знак “$” в красный.

**Вопрос 9**

**Какова роль данный отрывок кода?**

Console.WindowWidth = Width;

Console.WindowHeight = Height;

Console.CursorVisible = false;

Строки устанавливают размеры окна консоли равными значениями в виде Height и Width.

Console.CursorVisible = false – скрывает курсор.

**Вопрос 10**

**Какова роль данный отрывок кода?**

Snake snake = new Snake(10, 10, 3);

Food food = new Food(20, 10);

Random random = new Random();

Этот код создает экземпляры объектов

**Вопрос 11**

**Какова роль данной строки?**

int dx = 1, dy = 0;

Эти переменные хранят направление движения змейки.

**Вопрос 12**

**Какова роль данного цикла?**

while (true)

{

if (Console.KeyAvailable)

{

ConsoleKey key = Console.ReadKey(true).Key;

switch (key)

{

case ConsoleKey.UpArrow: if (dy != 1) { dx = 0; dy = -1; } break;

case ConsoleKey.DownArrow: if (dy != -1) { dx = 0; dy = 1; } break;

case ConsoleKey.LeftArrow: if (dx != 1) { dx = -1; dy = 0; } break;

case ConsoleKey.RightArrow: if (dx != -1) { dx = 1; dy = 0; } break;

}

}

Данный цикл отвечает за непрерывное выполнение кода, пока игра не будет завершена. Также он отображает движение змейки, после которой игра закончится.

snake.Move(dx, dy);

**Вопрос 13**

**Какова роль данного отрывка кода?**

if (snake.Body[0].X == food.X && snake.Body[0].Y == food.Y)

{

snake.Body.Add(new Point(snake.Body[snake.Body.Count - 1].X, snake.Body[snake.Body.Count - 1].Y));

food.X = random.Next(0, Width);

food.Y = random.Next(0, Height);

}

Console.Clear();

snake.Draw();

food.Draw();

Thread.Sleep(100);

Данный отрывок передает текущее направление dx и dy, также перемещает змейку на одну клетку в выбранном направлении.

Недостатки:

* Отсутствует счет
* При попадании в «хвост» змейки игра не заканчивается
* Зона игры не определена границами
* Во время игры окно с самой игрой моргает
* Невозможно проиграть
* Отсутствуют кнопки «Выход», «Рестарт», «Пауза»