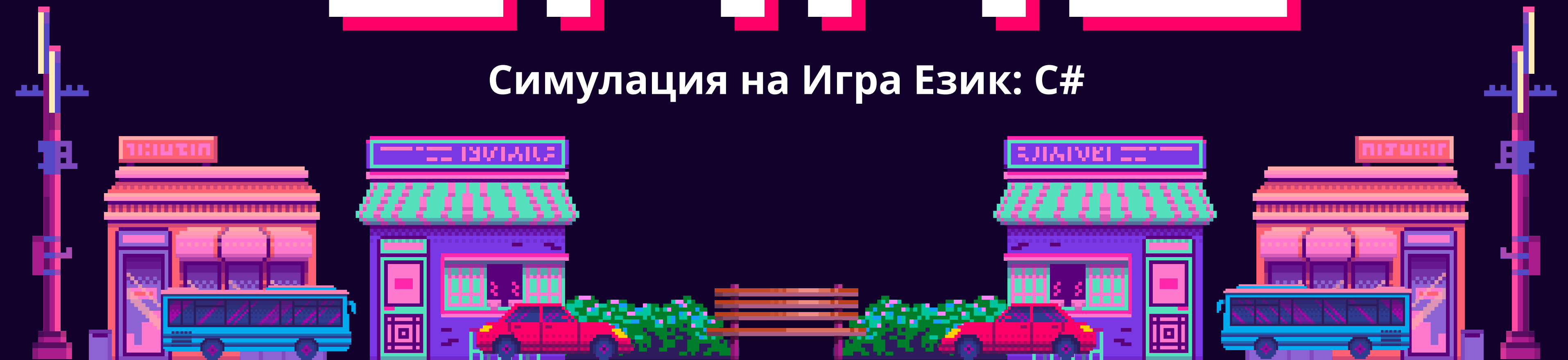


Изготвил: Калоян Ангелов 10г

GO4ME

Симулация на Игра Език: С#



ЦЕЛ НА ЗАДАЧАТА

Създаване на проста игра между
двама играчи

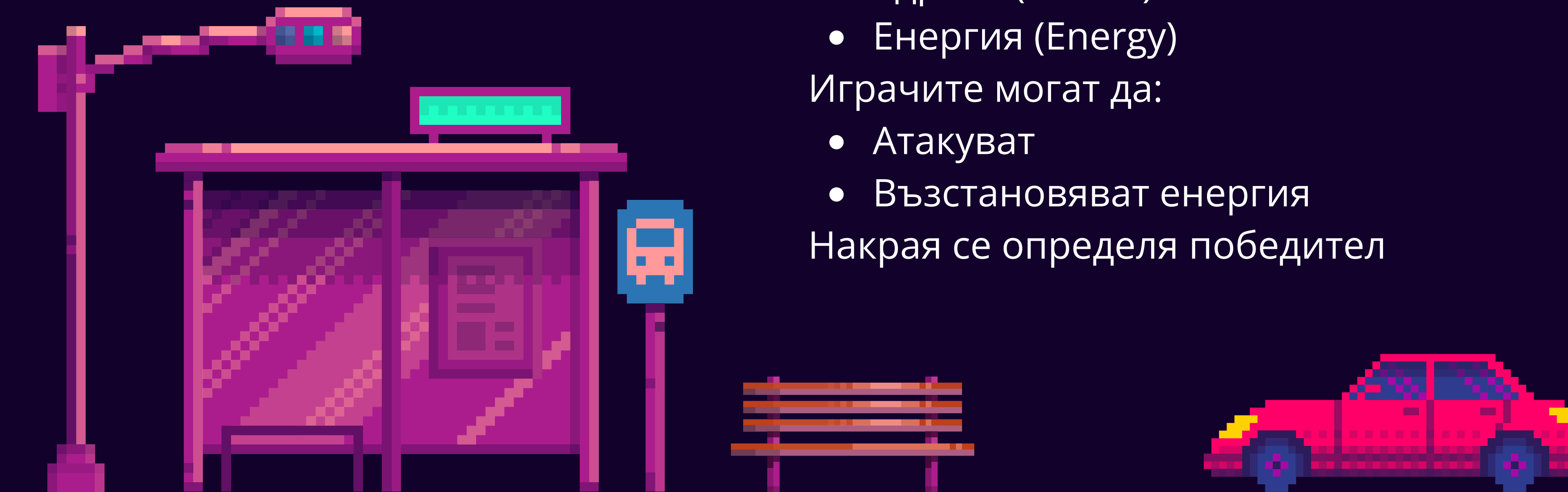
Всеки играч има:

- Име
- Здраве (Health)
- Енергия (Energy)

Играчите могат да:

- Атакуют
- Възстановяват енергия

Накрая се определя победител



ОСНОВЕНА ИДЕЯ

Програмата работи в цикъл

- Потребителят избира действия от меню
След всяко действие се показва текущото
състояние

Победителят се определя по формула:

$$\text{Power} = \text{Health} + \text{Energy}$$



СЪЗДАВАНЕ НА ИГРАЧИТЕ

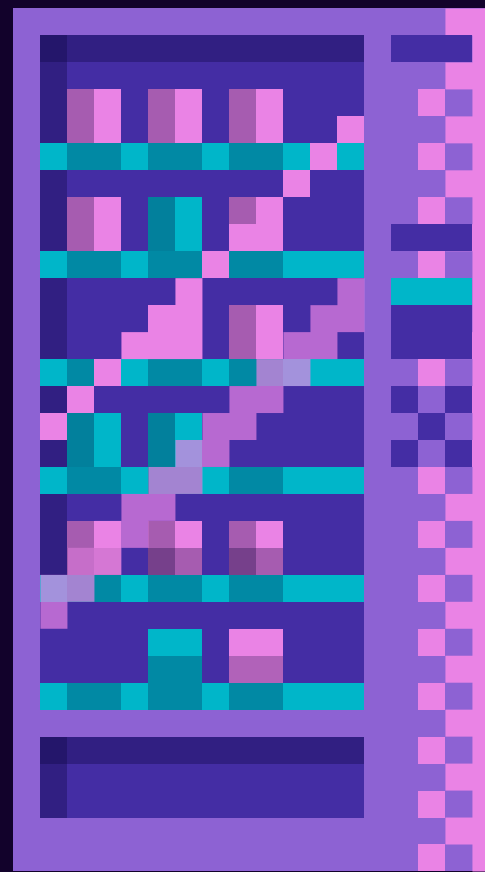
- Въвеждане на данни от конзолата:
 - Име
 - Здраве
 - Енергия
- Създаване на обекти от тип Player

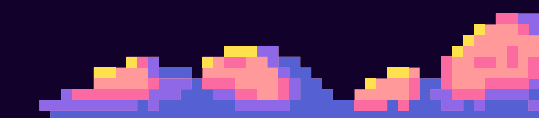
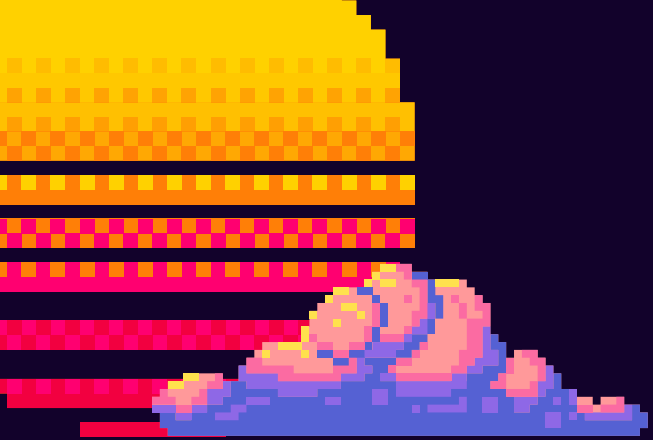
```
Player p1 = new Player(name1, health1, energy1);  
Player p2 = new Player(name2, health2, energy2);
```

НАЧАЛНО СЪСТОЯНИЕ

- След създаване на играчите:
 - Показваме началното им състояние

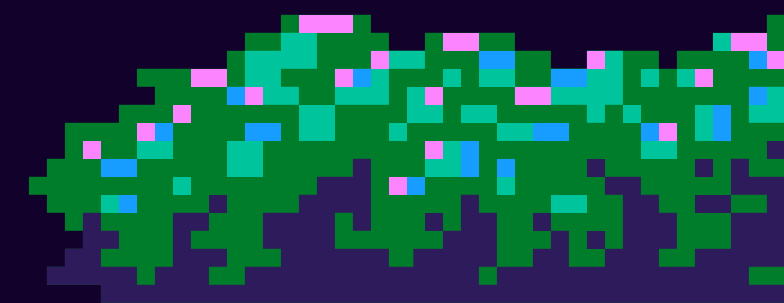
```
Console.WriteLine("\nInitial State:");  
Console.WriteLine(p1);  
Console.WriteLine(p2);
```





МЕНЮ С ДЕЙСТВИЕ

- Използва се while цикъл
- Потребителят избира действие:
 - a. Player 1 атакува Player 2
 - b. Player 2 атакува Player 1
 - c. Player 1 възстановява енергия
 - d. Player 2 възстановява енергия
 - e. Край на играта



ОБРАБОТКА НА ИЗБОРА (SWITCH)

- Използва се switch конструкция
- Всеки case изпълнява различно действие

```
switch (choice)
{
    case 1:
        Console.WriteLine("Enter damage amount: ");
        int dmg1 = int.Parse(Console.ReadLine());
        p2.TakeDamage(dmg1);
        break;

    case 2:
        Console.WriteLine("Enter damage amount: ");
        int dmg2 = int.Parse(Console.ReadLine());
        p1.TakeDamage(dmg2);
        break;

    case 3:
        Console.WriteLine("Enter energy amount to restore: ");
        int en1 = int.Parse(Console.ReadLine());
        p1.RestoreEnergy(en1);
        break;

    case 4:
        Console.WriteLine("Enter energy amount to restore: ");
        int en2 = int.Parse(Console.ReadLine());
        p2.RestoreEnergy(en2);
        break;

    case 5:
        running = false;
        break;

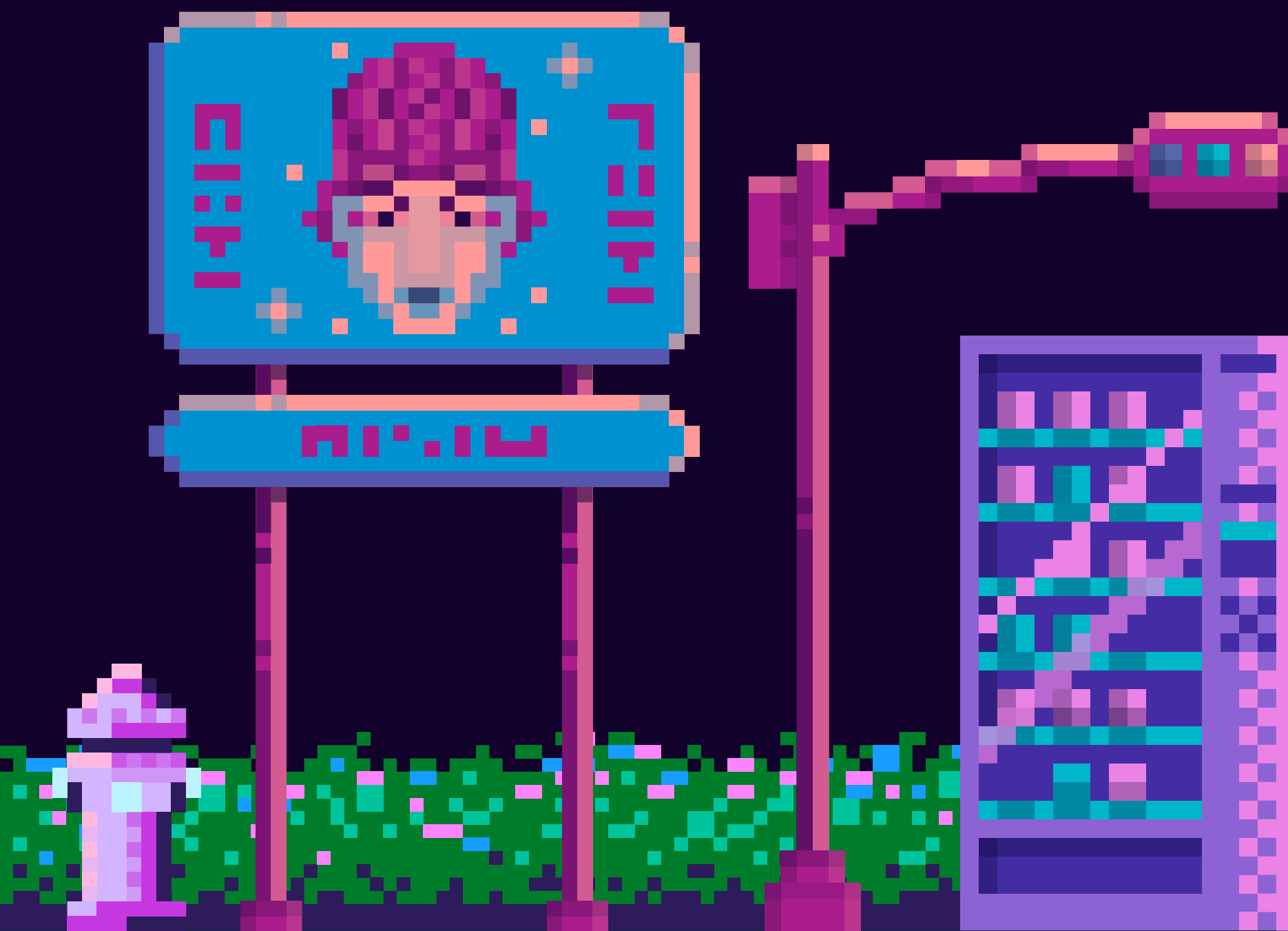
    default:
        Console.WriteLine("Invalid choice.");
        break;
}
```

КЛАС GAME RULES

- Статичен клас, който съдържа общи правила на играта
- Използва се за:
 - Максимално здраве
 - Ограничаване на стойности

```
public static int MaxHealth { get; set; } = 100;
```

- Метод ClampHealth(int value):
 - Не позволява Health да е под 0
 - Не позволява Health да надвишава MaxHealth



КЛАС PLAYER – СВОЙСТВА

- Представя един играч в играта
- Основни свойства:
- Name – име на играча
- Health – здраве
- Energy – енергия

```
9 references  
public string Name { get; set; }  
9 references  
public int Health { get; set; }  
9 references  
public int Energy { get; set; }
```



ОСНОВНИ МЕТОДИ И ИГРОВА ЛОГИКА

Клас Player съдържа основните действия в играта:

TakeDamage(int damage) – намалява здравето (минимум 0)

RestoreEnergy(int amount) – възстановява енергия (максимум 100)

Използва се **GameRules** за контрол на здравето

Методът **ToString()** показва текущото състояние на играча

```
public void TakeDamage(int damage)
{
    Health -= damage;
    if (Health < 0) Health = 0;

    Console.WriteLine($"{Name} took {damage} damage. Health: {Health}");
}

// ----- Метод 2: Възстановяване на енергия -----
2 references
public void RestoreEnergy(int amount)
{
    Energy += amount;
    if (Energy > 100) Energy = 100;

    Console.WriteLine($"{Name} restored {amount} energy. Energy: {Energy}");
}

0 references
public override string ToString()
{
    return $"{Name} | Health: {Health} | Energy: {Energy}";
}
```



ИЗЧИСЛАВАНЕ НА СИЛАТА И ПОБЕДИТЕЛ

- След всяко действие се изчислява силата на играчите:
- $\text{Power} = \text{Health} + \text{Energy}$
- Победител:
- По-голяма сила → победа
- Равни стойности → равенство

```
int p1Power = p1.Health + p1.Energy;  
int p2Power = p2.Health + p2.Energy;
```

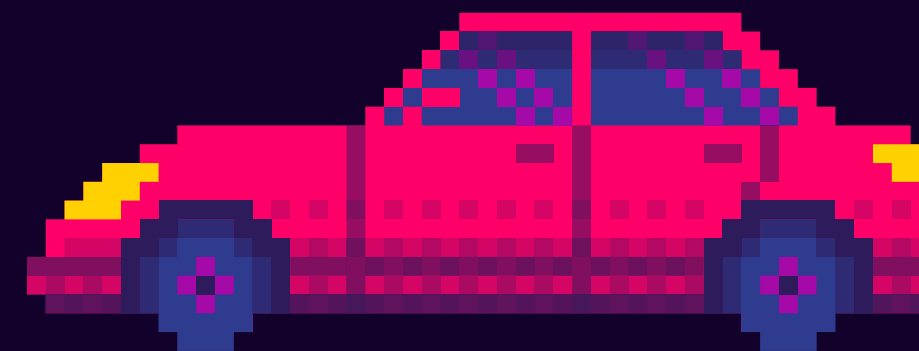
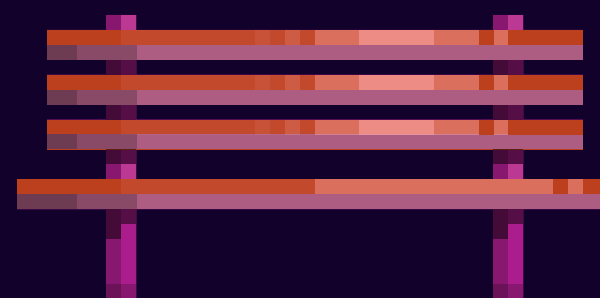
ЗАКЛЮЧЕНИЕ И ПОДДОБРЕНИЯ

Проектът демонстрира:

- Класове и обекти
- Статичен клас
- Условия и цикли
- Подходящ за начинаещи

Възможни подобрения:

- Край при 0 здраве
- Различни атаки
- Графичен интерфейс





БЛАГОДАРА ЗА
ВНИМАНИЕТО!