Отчет по практической работе №12

Студента группы №81

Лазо Константина Олеговича

**Используя практическую работу №11 дополнить экран игры движением 3-х монстров согласно описанию игры в WSR\_2\_MODULE.docx.  
Требуется реализовать все описанное выше в презентации.**

**Результат:**

Для начала я создал абстрактный класс Enemy. Движение монстров реализовал внутри класса игрового процесса.

При этом когда игрок бежит от монстров, он их обгоняет (медленнее всего обгоняет самого быстрого монстра), когда он бежит навстречу монстрам – монстры быстро приближаются. Монстры поворачиваются в сторону игрока. У монстров есть разница в скорости. Когда игрок стоит, монстры приближаются быстрее.



Класс Enemy:

import { Character } from '../abstract-main/character';

*/\*\**

*\* Абстрактный класс врага*

*\*/*

export class Enemy extends Character

{

    constructor(*game*, *x*, *y*)

    {

**super**(*game*, *x*, *y*)

**this**.y0 = *y*;

**this**.direction = 'left';

**this**.player = **this**.game.player; *// игрок*

*// цвет и высота полоски здоровья*

**this**.healthColor = '#14c94c';

**this**.healthHeight = 5;

*// перерыв между атакой в милисекундах*

**this**.attackCheck = 0;

**this**.timer = 0;

    }

*/\*\**

*\* Эффект "прыжка" у врага в движении*

*\* Прыжок опфределяет синусоида.*

*\* Первое число регулирует внутренний угол (Math.sin(x/n)), второе - выпуклость (bend)*

*\*/*

    jump()

    {

**this**.y = **this**.y0 - f(**this**.x);

        function f(*x*)

        {

            let bend = 48

            return Math.sin(*x* / 30) \* bend;

        }

    }

    init()

    {

        var rightWalkImg = **this**.rightWalkImg,

            leftWalkImg = **this**.leftWalkImg,

            direction = **this**.direction,

            x = **this**.x,

            y = **this**.y,

*// полоска здоровья*

            healthX = x + 50,

            healthY = y - 10

        if(direction == 'right')

        {

**this**.screen.fillRect(**this**.healthColor, healthX, healthY, **this**.health, **this**.healthHeight);

**this**.screen.drawImage(x, y, rightWalkImg, **this**.width, **this**.height);

        } else {

**this**.screen.fillRect(**this**.healthColor, healthX, healthY, **this**.health, **this**.healthHeight);

**this**.screen.drawImage(x, y, leftWalkImg, **this**.width, **this**.height);

        }

    }

}

Движение монстров из класса GameProcess:

*/\*\**

*\* Движение врага относительно игрока*

*\* "Враг" - абстрактный класс, так что условие работает для каждого созданного врага под этим классом*

*\* Если игрок бежит вправо, то он обгоняет врагов слева и приближает врага справа*

*\* Если игрок бежит влево, то наоборот*

*\* Примечание: до и после движения фона позиция игрока = "стоит" во избежание багов движения врагов*

*\*/*

    enemyMove()

    {

        for(let enemy in **this**.enemies)

        {

            enemy = **this**.enemies[enemy];

*// включает эффект прыжка у врага*

            enemy.jump();

*// игрок бежит ко врагам вправо*

            if(**this**.player.standing == false && **this**.player.direction == 'right')

            {

                enemy.x += enemy.step - (enemy.step + **this**.player.step);

*// враги слева гонятся за игроком*

                if(enemy.x + enemy.damageArea < **this**.player.x)

                {

                    enemy.x += enemy.step \* 1.5;

                    enemy.direction = 'right';

                }

            }

*//если игрок "бежит" влево*

            else if(**this**.player.standing == false && **this**.player.direction == 'left')

            {

                enemy.direction = 'left';

                enemy.x -= enemy.step \* 1.5 - **this**.player.step;

*// враги с другой стороны гонятся за игроком*

                if(enemy.x + enemy.damageArea < **this**.player.x)

                {

                    enemy.direction = 'right';

                    enemy.x += enemy.step \* 2;

                }

            }

            else if((enemy.x - enemy.damageArea > **this**.player.x) && **this**.player.standing == true)

            {

                enemy.direction = 'left';

                enemy.x -= enemy.step;

            }

            else if((enemy.x + enemy.damageArea < **this**.player.x) && **this**.player.standing == true)

            {

                enemy.direction = 'right';

                enemy.x += enemy.step;

            }

        }

    }