



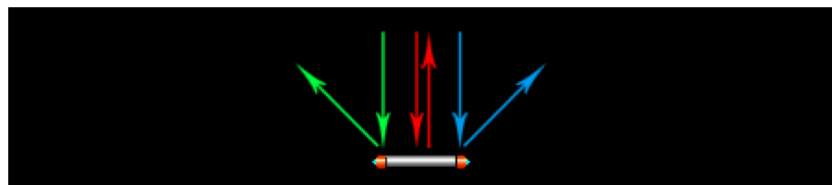
Versión final

1. Dirección aleatoria al lanzar la bola desde la pala (fácil)
2. Ajuste de dirección cuando rebota contra la pala (idea)

Mejora del movimiento de la bola

Colisión pala con bola

- ¿Cómo conseguir este efecto cuando la bola colisiona con la pala?



```
// ascii art:  
// -1  -0.5  0  0.5   1   <- x value  
// ===== <- paddle
```

```
(ballPos.x - paddlePos.x) / paddleWidth/2
```

Implementación

- Necesitamos
 - Posición de la bola + posición y ancho de la pala
- ¿Cómo/cuándo podemos conseguir esta información?
 - Al chocar la pala con la bola
- ¿Dónde implementamos el método que lo calcule?

```
public float HitFactor (Vector2 ballPos) {  
    // -1  -0.5  0  0.5  1  <- x value  
    // ===== <- paddle  
    return (ballPos.x - transform.position.x) / width;  
}
```

- ¿Qué es **width**?
 - Tiramos de la caja AABB (*bounds*) del *collider* y usamos ½ de su ancho (*extents*)

Motores de Videojuegos © 2019-20

Implementación

- Si la bola choca con la pala
 1. Le pide a la pala que calcule el *hitFactor*
 2. Construye un vector con ese valor para x (e y = 1)
 3. Lo normaliza
 4. Lo usa como vector de dirección
 5. Y lo multiplica por *speed* para respetar la velocidad configurada
- ¿Cómo preguntar si la bola ha chocado con la pala?
 - Sin usar *string typing* sino *duck typing*