

Colisões

```
public class PController05 : MonoBehaviour {
    public static PController05 instance;
    public PBody05[] bodys;
    public void CheckCollision (PBody05 b1){
        if (b1.isStatic) return;
        foreach (PBody05 b2 in bodys){
            if (b2 != b1){
                Vector3 d = b2.shape.position - b1.shape.position;
                float diff = (b2.shape.radius + b1.shape.radius) - d.magnitude;
                if (diff > 0) {
                    PCollision05 c = new PCollision05();
                    c.penetration = diff;
                    c.normal = d.normalized;
                    ResolvePenetration(c, b1, b2);
                }
            }
        }
    }
}
```

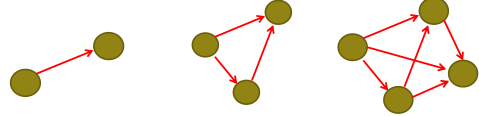
Colisões

Complexidade (O^2)

- Deve-se testar cada objeto contra todos os outros

Otimização

- Determinar objetos com chance de colisão (ex.: estáticos não se movem)
- Divisão de espaço (Árvore Binária, QuadTree, OctTree)
- Utilizar cálculo de colisão mais simples quando a distância é maior.



Colisões

Duas etapas:

Broad phase :

- Fase que detecta potencial colisão.
- Por questões de desempenho são usadas técnicas como:
 - Space partition
 - Uniform Grid
 - Quad trees (2D)
 - Octree (3D)
 - Sort and Sweep Algorithm (Bullet em bAxisSweep3)
 - Bounding volume.

Narrow Phase:

- fase que detecta colisão de fato.
- Maior precisão nos cálculos. Detectar ponto exato da colisão.

Colisões

```
void Start() {
    instance = this;
    GameObject[] list = GameObject.FindGameObjectsWithTag("Body");
    Debug.Log(list.Length);
    bodys = new PBody05[list.Length];
    for (int i = 0; i < list.Length; i++){
        bodys[i] = list[i].GetComponent<PBody05>();
    }
}

void ResolvePenetration (PCollision05 c, PBody05 b1, PBody05 b2){
    if (b2.isStatic){
        b1.shape.position += c.penetration * -c.normal;
    } else {
        b1.shape.position += c.penetration * 0.5f * -c.normal;
        b2.shape.position += c.penetration * 0.5f * c.normal;
    }
}
```

Engines de Física

Rigidbody x Objeto (Collider não Trigger)

Objeto que se move	Script	OnCollision	Visualmente
Rigidbody	velocity	OK	Para ao colidir (aSleep)
	Translate() ou position	OK	Não atravessa mas fica batendo
Rigidbody isKinematic	velocity	Não	Não se move
	Translate() ou position	Não	Atravessa o objeto
Objeto sem Rigidbody	Translate() ou position	Não	Atravessa o Rigidbody

Engines de Física

Rigidbody x Objeto (Collider Trigger)

Objeto que se move	Script	OnTrigger	Visualmente
Rigidbody	velocity	OK	Atravessa o objeto
	Translate() ou position	OK	Atravessa o objeto
Rigidbody isKinematic	velocity	OK	Não se move
	Translate() ou position	OK	Atravessa o objeto
Objeto sem Rigidbody	Translate() ou position	Não	Atravessa o Rigidbody

Colisões

Tarefa:

Implementar Colisão Esfera e Plano

- Detectar Colisão
- Resolver Interseção

