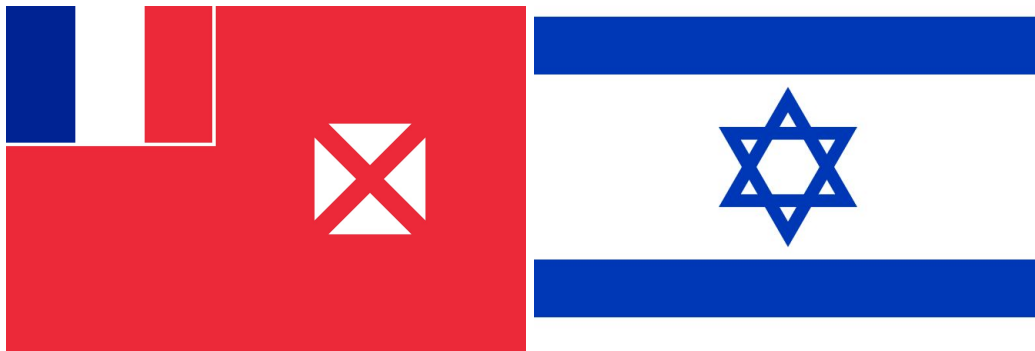


## Trabalho 2

**Alunos: Silvana Trindade e Maurício André Cinelli**

Descrição: *Desenhar a bandeira de algum país e mostre-a tremulando num mastro.*

As bandeiras seleccionadas foram de *Wallis e Fortuna*, e também a de *Israel*, podendo ser observadas na imagem abaixo.



Para obtermos uma simulação mais realista de uma bandeira ao vento, fizemos uma simulação de tecido. Para entendermos como funciona, buscamos várias informações na internet e livros. No artigo do Wikipedia, são mostrados dois métodos, porém o segundo método é o que também encontramos nos livros.

A maioria dos modelos são baseados em partículas de massa conectadas em uma forma de malha. Física Newtoniana é usada para modelar cada partícula, mais especificamente a lei básica de movimento (segunda Lei de Newton).

$$\vec{F} = m\vec{a}$$

A simulação se dá pela massa e interconexões entre as partículas, chamadas de restrições (*constraints*) ou molas (*springs*). Como desejamos calcular quanto forças como gravidade e vento afetam as partículas, podemos calcular a aceleração delas (força dividida pela massa,

segundo Lei de Newton) e depois as suas novas posições, através de uma integral, chamada de Método de Verlet, específica para a integração da segunda lei de Newton. O método de Verlet utiliza a posição anterior do objeto para calcular a próxima posição do objeto, de forma a calcular a aceleração, e posteriormente convertê-la na posição de fato.

## Código

O código foi todo estruturado em classes. A seguir temos as descrições de cada classe de forma resumida, focando no seu objetivo.

### Vector3

A Classe *Vector3* foi baseada no código da classe apresentada no Capítulo 6 da referência [2]. É uma classe com o intuito de facilitar nos cálculos de vetores e pontos, sem ter que utilizar uma biblioteca mais completa. A classe contempla operações básicas entre vetores, como por exemplo a soma e subtração, e de vetores com escalares, como multiplicação e divisão. Faz também o cálculo da magnitude, normal, produto interno e externo.

### Restrição

A Classe *Restrição* tem como objetivo fazer com que as partículas do tecido voltem ao seu estado normal depois de serem movimentadas. Porém, logo depois de fazê-las voltarem, o código da próxima iteração já é chamado. Com isso, o tecido pode apenas esticar um pouco ou encolher, mas não muito além do que a restrição mantém. Como restringe o movimento das partículas, alterando os parâmetros, seria equivalente a ter um tecido mais rígido ou mais flexível.

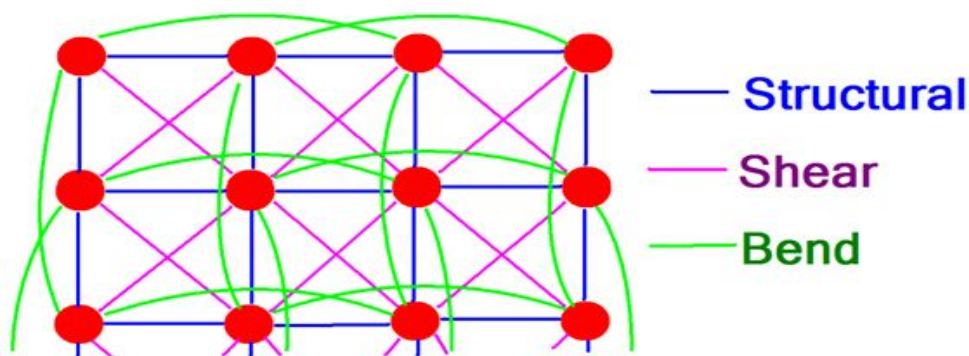
### Partícula

A Classe *Partícula* representa um ponto, mas que permite ser adicionadas forças, como gravidade e vento. É aqui onde é utilizado o método de Verlet para calcular a próxima posição da partícula a cada fatia de tempo (frame).

Nesta classe também há uma flag que indica se a partícula está fixa ou não. Isso nos permite dar o efeito de que a bandeira está presa ao mastro, já que na verdade, apenas não calculamos a próxima posição.

## Tecido

Por fim, a classe tecido, é somente um conjunto de partículas e restrições. O objetivo dessa classe é coordenar os frames das partículas, e montar as restrições de forma a manter algumas propriedades do tecido:



As linhas azuis representam as restrições estruturais, estas previnem que as partículas fiquem muito distantes uma da outra, ou que encolham em um ponto. Porém, não previnem que o tecido dobre sobre si mesmo, formando um vinco infinitamente abrupto. Para limitar como o tecido deve dobrar, são adicionadas restrições com partículas de distância 2, que são as linhas em verde. Porém, há uma situação que ainda pode acontecer, que é o *shear*, ou seja, o tecido pode ser deformado e formar um losango. Para isso são adicionadas restrições com as partículas na diagonal, como poder ser vista as linhas em rosa. A classe tecido também é a responsável por de fato desenhar os triângulos, com base nas novas posições das partículas, e pintá-las com base na matriz de cores que estabelecemos para pintar a bandeira.

## Resultados Obtidos

Bandeira de Wallis e Fortuna:



Bandeira de Israel:



## Referências

- [1] Wikipedia. Cloth Modeling. [https://en.wikipedia.org/wiki/Cloth\\_modeling](https://en.wikipedia.org/wiki/Cloth_modeling)
- [2] 3D Math Primer for Graphics and Game Development, 2002.
- [3] Mathematics for 3D Game Programming and Computer Graphics, Third Edition, 2012.

*OBS:* Livros disponíveis na biblioteca do campus.