

Segue a mini-especificação do que deve ser feito no exercício 1:

#### Classe Planeta

Deve ter os atributos:

Sprite \*sprite;

float x, y;

float hitpoints;

Deve ter os métodos:

Planet(Sprite \*sprite, float hitpoints, float x, float y);

Não devem utilizar SDLBase::loadImage mais de uma vez para o sprite do planeta, carregar a imagem apenas uma vez no gameManager e passar a referência.

void render();

Renderizar o planeta, de acordo com suas posições x e y

#### Alterações no GameManager

Novos atributos:

Sprite \*planetSprite;

std::vector<Planet\*> planetArray;

Novos métodos:

void addPlanet();

- Os planetas devem ter de 1-20 hit points e ser criados em uma posição aleatória DENTRO da área da tela.

- Os planetas devem ser guardados no vector.

void checkPlanets();

- Checar se algum planeta tem vida menor ou igual a 0. Se tiver, remover o planeta do vetor

void processEvents();

- Método disponível no moodle, deve ser usado sem alterações.

- Esse método faz o controle do input do teclado: Chama o método addPlanet, testa colisão do mouse com os planetas do vetor e aplica o dano, quando necessário.

#### Alterações no método run (Main Game Loop):

- Input: Chamar o método processEvents();

- Update: Chamar o método checkPlanets();

- Render: Mostrar o BackGround e todos os planetas e atualizar a tela.

Os exercícios práticos não tem um critério de correção definido minuciosamente, eles devem apenas funcionar corretamente e respeitar a especificação.

Vale lembrar que esse exercício (e todos os próximos estarão incorporados aos próximos trabalhos, onde terão uma avaliação específica).

Ou seja, no fim, seu exercício precisa de:

Planetas sendo Criados, destruídos e renderizados de maneira correta. (10)

O não uso correto de Orientação a Objetos ou não respeitar a especificação pode e acarretará em perda de pontos!

Apenas lembrando a Regra de Nomenclatura (teve gente que errou no trabalho, PRESTEM ATENÇÃO):

O aluno deve nomear seu arquivo .zip contendo todos os arquivos utilizados no trabalho com

<matrícula>\_<tr/ex><nro>.zip, onde:

<matrícula> é a matrícula do aluno sem barra, <tr/ex> deve ser usado para identificar se é um trabalho ou exercício e <nro> é o número do trabalho. (Ex: 0012345\_ex01.zip)