

## Trabalho (Etapa 2)

As atividades descritas a seguir devem seguir as seguintes restrições:

- Todas devem ser implementadas em grupos de até três;
- As atividades devem ser implementadas utilizando a API da OpenGL, sendo as bibliotecas gl, glu, glut as únicas que podem ser utilizadas.
- As linguagens permitidas são C e C++.
- Para submissão, aqueles que optarem por utilizar Windows devem compactar todo o código-fonte como um arquivo “.zip”, incluindo executável. Os que optarem por sistemas operacionais baseados em UNIX também devem enviar todo o código-fonte compactado, acompanhado de um Makefile. Em ambos os casos, um arquivo com extensão “.txt” deve ser incluído, especificando os nomes dos integrantes do grupo.
- Caso haja plágio ou trabalhos muito similares, os grupos envolvidos receberão nota 0.

### Contextualização do trabalho



Na primeira etapa do trabalho, foi requisitada a implementação de um jogo 2D, que consistia na construção de dois personagens com a formas de pinguins e a modelagem de personagens secundários, no formato de peixes e aves. Um dos pinguins se movimentava controlado pelas teclas do teclado com objetivo de capturar peixes no lago e levar até o seu filho, sem que este fosse atingido pelo o pássaro Petrel-Gigante, assumindo uma cena em um espaço bidimensional.

Para fixação dos conteúdos vistos até então, expandiremos a implementação do movimento do pinguim para um espaço tridimensional. Nesta versão do jogo:

- **Os pinguins (mãe e filho) estão sob uma placa de gelo;**
- **O filho permanecerá parado no centro dessa placa e a mãe irá em busca dos peixes que irão aparecer de forma aleatória sob o gelo;**
- Para capturar um peixe o pinguim deve colidir com ele. Para representar que o pinguim já capturou um peixe, este deve ser desenhado preso ao bico do pinguim;
- Depois de ter um peixe preso a seu bico, o pinguim não consegue capturar novos peixes, até que o entregue ao filhote;
- **A pinguim mãe deve tomar cuidado para não cair nos buracos que aparecem (também aleatoriamente) na placa de gelo;**

- O alimento dá energia ao filhote para que este viva por 1 minuto. Passado este tempo ele morre e o jogo termina;
- Para alimentar o filhote o pinguim precisa apenas se aproximar dele, carregando um peixe. Com este movimento, o peixe desaparece da boca do pinguim;
- O jogo deve durar no máximo 5 minutos. Se neste tempo o filhote permanecer vivo, o jogador ganha o jogo.

## Questões propostas

1. Implemente dois atores que tenham a forma de pinguins. Modele o pinguim utilizando esferas, elipsóides e cones.

Alguns exemplos são apresentados a seguir.



2. Modele os peixes que irão aparecer (aleatoriamente) sob o gelo utilizando elipsóides e cones.

3. Permita que, ao utilizar as setas do teclado, a pinguim mãe se movimente de maneira correspondente. No caso das setas para cima e para baixo, o pinguim deve se deslocar ou para frente ou para trás, seguindo o eixo que ela está direcionada. Já com as setas da esquerda e da direita, o pinguim deve rotacionar em torno do seu próprio eixo.

**Questão bônus** - Durante o seu deslocamento o pinguim deve movimentar as asas para frente e para trás, simulando o andar.

4. Implemente o aparecimento aleatório de buracos **no caminho do pinguim** sob o gelo.

5. Implemente quatro janelas para visualização da cena, definindo quatro *viewports* distintos, obedecendo às seguintes restrições:

- Deve haver pelo menos uma janela em que a câmera está “posicionada” acima da cena, no eixo Y;
- Deve haver pelo menos uma janela em que a câmera está “posicionada” do lado da cena, no eixo X;
- Deve haver pelo menos uma janela em que a câmera está “posicionada” de frente para a cena, no eixo Z;
- Deve haver pelo menos uma janela em que a câmera está “posicionada” em uma posição livre, como por exemplo nas coordenadas (3.0, 2.0, 10.0).

4. Manipule a função `gluLookAt(...)` de maneira a sempre manter a pinguim mãe no centro da porção visível da cena. Isso significa que a câmera deve ter como ponto focal central a aranha, em todas as janelas do programa.

6. Insira três texturas na cena, dando um maior grau de realismo nos personagens (pinguins e peixes), placa de gelo e buracos e ao redor da cena.

- A escolha das imagens que serão utilizadas como texturas é livre.

6. O grupo também pode utilizar uma ou mais fontes de iluminação para a cena (OPCIONAL).

**Questão bônus (pesquise sobre os conceitos envolvidos na resolução dessa questão):**

No lugar de desenhar um “chão” para a movimentação da aranha, utiliza-se o conceito de *skybox*<sup>1</sup>, em que seis imagens distintas devem ser dispostas de maneira a formar um cubo. Para essa textura, utilize somente *cubemap* para definição dos parâmetros (cujas constantes são definidas por `GL_TEXTURE_CUBE_MAP`).

<sup>1</sup>[https://en.wikipedia.org/wiki/Skybox\\_\(video\\_games\)](https://en.wikipedia.org/wiki/Skybox_(video_games))