

## Projeto sobre Computação Gráfica 3D

Desenvolva uma aplicação em Java 3D que demonstre a matéria estudada nos capítulos de 5 a 12.

De preferência desenvolva uma aplicação com um tema específico. Veja os temas sugeridos no final deste documento. Em alternativa, pode optar por desenvolver uma cena 3D abstrata, sem tema específico.

De modo a integrar os tópicos da matéria no miniprojeto, pode optar por integrar alguns na implementação do tema e os que não for possível ou difícil, pode integrá-los num *splash screen* da aplicação, por exemplo.

A nota do miniprojeto tem um peso de 20% na nota final. A data limite para entregar o miniprojeto via moodle, é dia 15 de janeiro de 2023. A avaliação do miniprojeto será realizada com base na apresentação realizada pelo aluno em datas a combinar.

A nota do miniprojeto tem as seguintes componentes e pesos:

- Complexidade do tema (20%)
- Quantidade da matéria estudada que foi aplicada no miniprojeto, de acordo com a tabela 1 (80%).

Para obter a percentagem máxima de cada item da tabela 1, deve apresentar exemplos diferentes dos realizados nas aulas.

TODO:  
Adicionar mais lights e rever

**Tabela 1 - Matéria estudada nos capítulos 5 a 12**

Background, Bounds e Capability bits	3%
Primitivas do Java 3D	4%
Geometrias por medida (diferentes das usadas nas aulas)	10%
Geometrias avançadas com GeometryInfo (diferente das usadas nas aulas)	4%
Fontes e Texto 3D	2%
Aparência com configuração de atributos "simples"	5%
Aparência com configuração de materiais	5%
Transformações Geométricas	5%
Configuração da vista	5%
Picking	5%
Iluminação	5%
Texturas 2D	10%
Behaviors de interação com objectos (MouseBehavior, KeyNavigatorBehavior, etc)	2%
Behaviors de interação com a vista (OrbitBehavior)	2%
Behaviors de animação predefinidos (Interpolator)	5%
Behaviors personalizados (de animação e/ou interação e diferentes dos usados nas aulas)	10%
Outros behaviors predefinidos (Morph, LOD, Billboard)	5%

Técnicas avançadas ( <u>Som 3D</u> , <u>Simulação de sombra</u> , <u>Geometrias dinâmicas</u> , <u>renderização off-screen</u> , <u>Texturas 3D</u> )	8%
GUI Swing/AWT → <i>Autos</i>	5% ✓

## Project about 3D Computer Graphics

Develop a Java 3D application that demonstrates the topics about 3D Computer Graphics studied in chapters 5-12.

Preferably develop an application with a specific theme. See suggested themes at the end of this document. Alternatively, you can choose to develop an abstract 3D scene, without a specific theme.

In order to integrate the topics in the mini-project, you can choose to integrate some in the implementation of the theme itself and those that are not possible or difficult to do so, you can integrate them into a splash screen of the application, for example.

The mini-project grade has a weight of 20% in the final grade. The deadline for submitting the project via moodle, is January 15, 2023, The evaluation of this mini-project will be based on the presentation made by the student on dates to be agreed with the professor.

The mini-project grade has the following components and weights:

- Theme complexity (20%)
- Quantity of the studied material that was applied in the project, according to table 1 (80%).

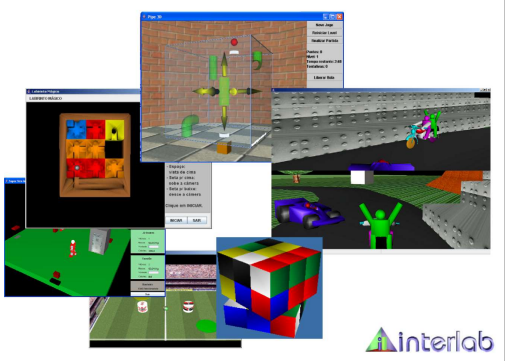
To obtain the maximum percentage of the items in table 1, you must present examples different from those implemented in class.

**Table 1 - 3D Computer Graphics topics of chapters 5-12**

Background, Bounds and Capability bits	3%
Java 3D Primitives	4%
Custom geometries (different from those used in classes)	10%
Advanced Geometries with GeometryInfo (different from those used in classes)	4%
Fonts and 3D Text	2%
Appearance with "simple" attributes	5%
Appearance with material configuration	5%
Geometric transformations	5%
View configuration	5%
Picking	5%
Lighting	5%
2D Textures	10%

Interaction Behaviors applied to objects (MouseBehavior, KeyNavigatorBehavior, etc)	2%
Interaction objects applied to the view (OrbitBehavior)	2%
Predefined Animation Behaviors (Interpolator)	5%
Custom Behaviors (animation and/or interaction and different from those used in classes)	10%
Other predefined behaviors (Morph, LOD, Billboard)	5%
Advanced technics (3D Sound, Shadow simulation, Dynamic geometries, off-screen renderization, 3D Textures)	8%
GUI Swing/AWT	5%

### Lista de possíveis temas / List of possible themes

<b>1</b>	<p><b>Jogo simples de tabuleiro</b></p> <p>Crie uma aplicação que implemente um jogo simples onde o jogador move peças com o teclado e/ou rato de modo a atingir determinado objetivo.</p> <p><b>Simple board game</b></p> <p>Create an application that implements a simple game where the player moves pieces with the keyboard and/or mouse in order to reach a certain objective.</p>
<b>2</b>	<p><b>Jogo simples com animação e interação</b></p> <p>Crie uma aplicação que implemente um jogo simples onde o jogador controla o movimento de uma personagem com o teclado e/ou rato, de modo a cumprir determinada missão e onde existem obstáculos estáticos e dinâmicos.</p> <p><b>Simple game with animation and interaction</b></p> <p>Create an application that implements a simple game where the player controls the movement of a character with the keyboard and/or mouse, in order to fulfill a certain mission and where there are static and dynamic obstacles.</p> 
<b>3</b>	<p><b>Visualização gráfica criativa de dados</b></p> <p>Implemente um ou mais gráficos/diagramas descritos no site <a href="https://datavizproject.com/">https://datavizproject.com/</a> na forma de funções ou classes e demonstre numa aplicação, o seu uso na visualização de conjuntos de dados á sua escolha (de</p>

preferência dados reais). Escolha um gráfico/diagrama com alguma complexidade e diferentes do que foi feito nas aulas.

### Creative graphical visualization of data

Implement one or more graphs/diagrams described on the website <https://datavizproject.com/> in the form of functions or classes and demonstrate in an application, their use in the visualization of datasets of your choice (preferably real data). Choose a graph/diagram with some complexity and different from what was done in class.



## 5 Visualização animada de algoritmos

Implemente uma aplicação que demonstre didaticamente o funcionamento de um ou mais algoritmos. Inspire-se nos exemplos do site <https://tamimehsan.github.io/AlgorithmVisualizer/#/>

### Animated visualization of algorithms.

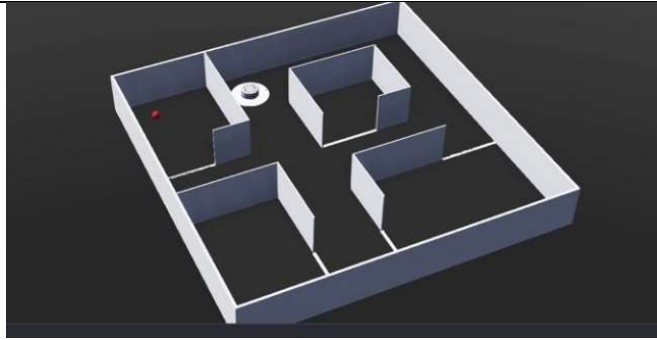
Implement an application that didactically demonstrate the operation of one or more algorithms. Get inspired by the site's examples <https://tamimehsan.github.io/AlgorithmVisualizer/#/>

## 6 Demonstração do concurso do Robô bombeiro

Implemente uma aplicação que demonstre uma prova típica do Concurso Robô Bombeiro, descrita na disciplina de Robótica do 2º ano. Opte por uma demonstração com animações predefinidas ou implemente um simulador simples com um robô a mover-se autonomamente.

### Demonstration of the Firefighter Robot contest

Implement an application that demonstrates a typical proof of the Firefighter Robot contest, described in the 2nd year Robotics class. Choose a demo with predefined animations or implement a simple simulator with a robot moving autonomously.



7

### **Simulador de braço robótico**

Implemente uma aplicação que simule o funcionamento de um braço robótico. Opte por implementar uma demonstração com animações predefinidas, movimentos controlados pelo teclado ou implemente um simulador simples, mas realista, baseado num modelo matemático da cinemática do robô.

#### **Robotic arm simulator**

Implement an application that simulates the operation of a robotic arm. Choose to implement a demo with predefined animations, with keyboard-controlled movements, or implement a simple but realistic simulator based on a mathematical model of the kinematics of the robot.

