


Tópicos Especiais em Linguagens de Programação

Apresentação da Disciplina

Edirlei Soares de Lima
<edirlei@iprj.uerj.br>



Objetivos da Disciplina

- Apresentar os fundamentos de **jogos eletrônicos**, **game design** e as **técnicas de programação para o desenvolvimento de jogos 2D e 3D**.
- Ao final do curso, o aluno deverá:
 - Compreender os conceitos fundamentais para a elaboração de projetos de jogos eletrônicos;
 - Conhecer e saber aplicar as técnicas fundamentais para o desenvolvimento e implementação de jogos eletrônicos.

Jogos Eletrônicos

Design de Jogos

+

Programação de Jogos 2D/3D

+

Modelagem e Animação 2D/3D



Ementa da Disciplina

- **Parte 1 – Jogos Eletrônicos e Game Design**
 - Fundamentos de jogos eletrônicos;
 - Projeto de jogos eletrônicos;
 - Concepção e criação;
 - Equipe de desenvolvimento;
 - A indústria de jogos;
 - Game design document;
 - Formato;
 - Gameplay;
 - Personagens;
 - Câmeras;
 - Interfaces.

Ementa da Disciplina

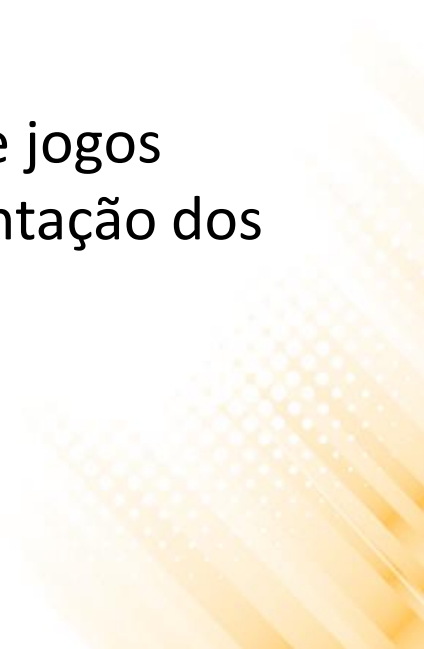
- **Parte 2 – Técnicas para Jogos 2D**
 - Sprites e animação 2D;
 - Introdução à linguagem Lua e ao framework Löve;
 - Técnicas de programação de jogos 2D:
 - Lógica e estrutura de um jogo;
 - Programação para ambientes gráficos;
 - Mecanismos de interação;
 - Estruturação e representação de dados;
 - Detecção de colisão;
 - Simulação Física.

Ementa da Disciplina

- **Parte 3 – Técnicas para Jogos 3D**

- Modelagem e animação 3D;
- Introdução à game engine Unity3D;
 - Arquitetura de game engines;
 - Interface e ambiente de desenvolvimento;
 - Importação de recursos (modelos, animações, texturas, áudio);
 - Criação de ambientes 3D;
 - Programação por scripts.
- Técnicas de inteligência artificial:
 - Conceitos e estado da arte;
 - Waypoints e pathfinding;
 - Máquinas de estados finitos.

Esquema de Aulas

- **Aulas teóricas e praticas:**
 - **Aulas teóricas** sobre fundamentos de jogos eletrônicos, game design, tecnologias, programação e técnicas de desenvolvimento;
 - **Aulas praticas** para a elaboração de projetos de jogos eletrônicos, resolução de exercícios e implementação dos projetos;
- 

Critério de Avaliação

- **Avaliação Prática:**
 - Trabalhos e exercícios desenvolvidos em grupo ou individualmente;
 - Apresentação em aula;
- **Avaliação Teórica:**
 - Prova teórica envolvendo o conteúdo teórico e prático apresentado durante as aulas;

Critério de Avaliação

- **Trabalhos:**

- T1: game design document;
- T2: desenvolvimento de um jogo 2D;
- T3: desenvolvimento de um jogo 3D;
- $MT = (T1+T2+T3)/3$

- **Prova:**

- PT: prova teórica individual envolvendo todo o conteúdo do curso;

- **$MP = MT * (PT/10)$**


Critério de Avaliação

- Se a frequência nas aulas for $< 75\%$ o aluno será REPROVADO POR FALTA;
- Se a frequência nas aulas for $\geq 75\%$, então:
 - Se $MP \geq 7.0$, o aluno será aprovado e $MF = MP$;
 - Se $MP < 4.0$, o aluno será reprovado e $MF = MP$;
 - Se $MP < 7.0$ e $MP \geq 4.0$, o aluno irá para o exame final e então:
 - $MF = (MP + PF)/2$;
 - Se $MF < 5.0$ o aluno será reprovado;
 - Se $MF \geq 5.0$ o aluno será aprovado;

Avaliação

- **Trabalhos desenvolvidos em equipe:**
 - 3 ou 4 alunos por equipe;
 - Os grupos serão definidos nas próximas aulas;
- **Avaliação individual baseada nos trabalhos desenvolvidos e na apresentação:**
 - Desenvolvimento;
 - Participação;
 - Apresentação;
 - Questionamentos;
 - Notas individuais;


Pré-Requisito

- Introdução à Programação
 - Gostar de Jogos
- 

Controle de Turma

- **Presença obrigatória!!!**
 - Chamada em qualquer momento da aula;
 - Alunos com menos de 75% de presença serão reprovados automaticamente e não poderão fazer prova final (independente da nota);
 - Não será permitido trocar de turma ou frequentar as aulas de outra turma;
 - Não será aberta nenhuma exceção!
- **Projetos desenvolvidos em equipe:**
 - Todos os membros da equipe devem participar ativamente do desenvolvimento dos projetos;
 - Não participar do desenvolvimento dos projetos resultará em redução individual da nota;

Aulas Praticas

- Nas aulas práticas usar apenas os softwares da disciplina;
 - Se for possível, trazer notebook pessoal para as aulas práticas;
 - Se for usar os computadores do laboratório, salve os trabalhos no pendrive com regularidade;
 - Verificar regularmente o site da disciplina;
 - Prática extraclasse constante (mínimo de 4h/semana);
- 

Software

- Os softwares que serão utilizados nesta disciplina são:
 - Löve - <https://love2d.org/>
 - ZeroBrane Studio - <http://studio.zerobrane.com/>
 - GIMP - <http://www.gimp.org/>
 - Unity 3D - <http://unity3d.com/>

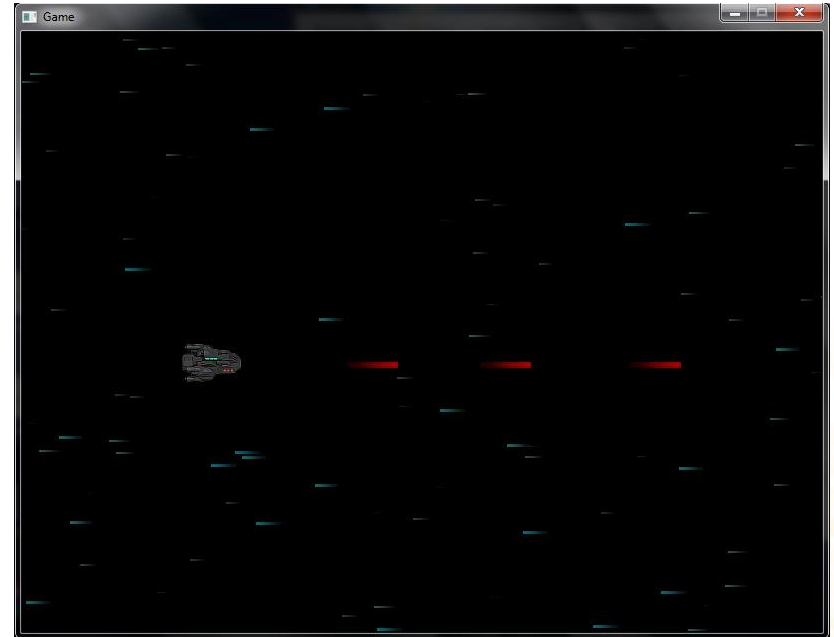
Exemplo 2D

```
...
function love.draw()
  for i = 1, config.numStars do
    love.graphics.setColor(255 - stars[i].speed,
                          255 - stars[i].speed/2, 150,
                          stars[i].speed*0.9)
    love.graphics.draw(star, stars[i].x, stars[i].y,
                      0, stars[i].speed/255 + 0.55)
  end

  love.graphics.setColor(255, 255, 255)

  if spaceship.isBoosted == false then
    love.graphics.draw(spaceship.images.normal,
                      spaceship.x, spaceship.y)
  else
    love.graphics.draw(spaceship.images.boosted,
                      spaceship.x, spaceship.y)
  end

  for i = 1, totalLaserBeams do
    love.graphics.draw(laser.images.default,
                      lasers[i].x, lasers[i].y)
  end
end
...
```

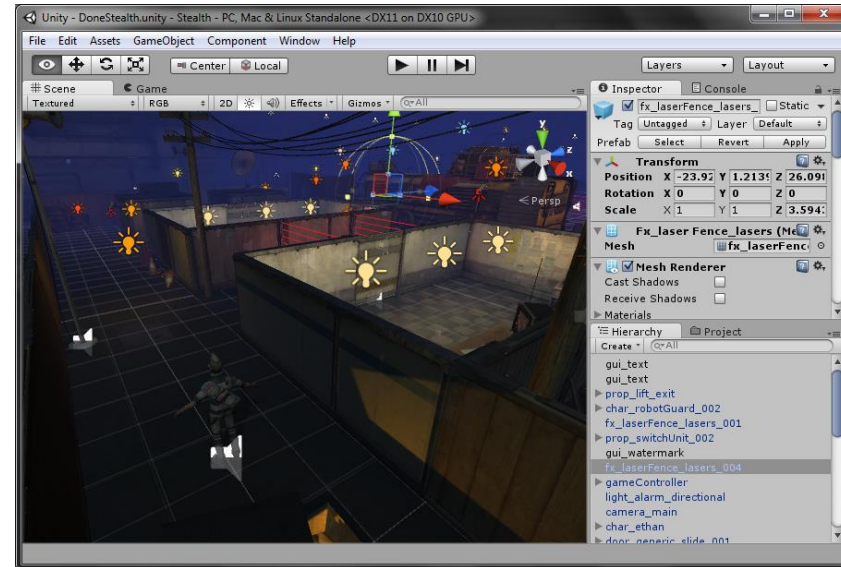


Exemplo 3D

```
...
void Update ()
{
    if(health <= 0f)
    {
        if(!playerDead)
            PlayerDying();
        else
        {
            PlayerDead();
            LevelReset();
        }
    }
}

void PlayerDying ()
{
    playerDead = true;
    anim.SetBool(hash.deadBool, playerDead);
    AudioSource.PlayClipAtPoint(deathClip,
                                transform.position);
}

void PlayerDead ()
{
    anim.SetFloat(hash.speedFloat, 0f);
    playerMovement.enabled = false;
    audio.Stop();
}
...
```



Material das Aulas

- **Página do Curso:**
 - www.inf.puc-rio.br/~elima/jogos/
- **Contato:**
 - edirlei@iprj.uerj.br

Bibliografia Principal

- Rogers, S. **Level Up!: The Guide to Great Video Game Design**; Wiley, 2010.
- Ierusalimschy, R. **Programando em Lua**, Terceira Edição, LTC, 2015.
- Blackman, S. **Beginning 3D Game Development with Unity 4**, Apress, 2013.



Bibliografia Complementar

- Fullerton, T., Swain, C., Hoffman, S. **Game Design Workshop: Designing, Prototyping and Playtesting Games**, CMP Books, 2004.
- Rollings, A., Morris, D. **Game Architecture and Design**, The Coriolis Group, 2000.
- Hocking, J. **Unity in Action: Multiplatform Game Development in C# with Unity 5**, Manning Publications, 2015.
- Millington, I., Funge, J. **Artificial Intelligence for Games**, Morgan Kaufmann, 2009.