

**DCC192**

2025/1

UF  G

# Desenvolvimento de Jogos Digitais

## A1: Introdução

Prof. Lucas N. Ferreira

# Professor



## Lucas N. Ferreira

Professor do Departamento de Ciência da Computação da UFMG  
Dr. em Ciência da Computação pela University of California, Santa Cruz

## Inteligência Artificial & Criatividade

Geração Musical, Geração Procedural de Conteúdo, Criatividade Computacional, Game AI

## Contato

Email - [lferreira@dcc.ufmg.br](mailto:lferreira@dcc.ufmg.br)

Página Pessoal: <http://www.dcc.ufmg.br/~lferreira>

# Estudantes



Meu nome é ...

Sou aluno do ... ano

Tenho experiência com desenvolvimento de jogos ...

Gostaria de desenvolver um jogo de ...

# Como se tornar um desenvolvedor de jogos?

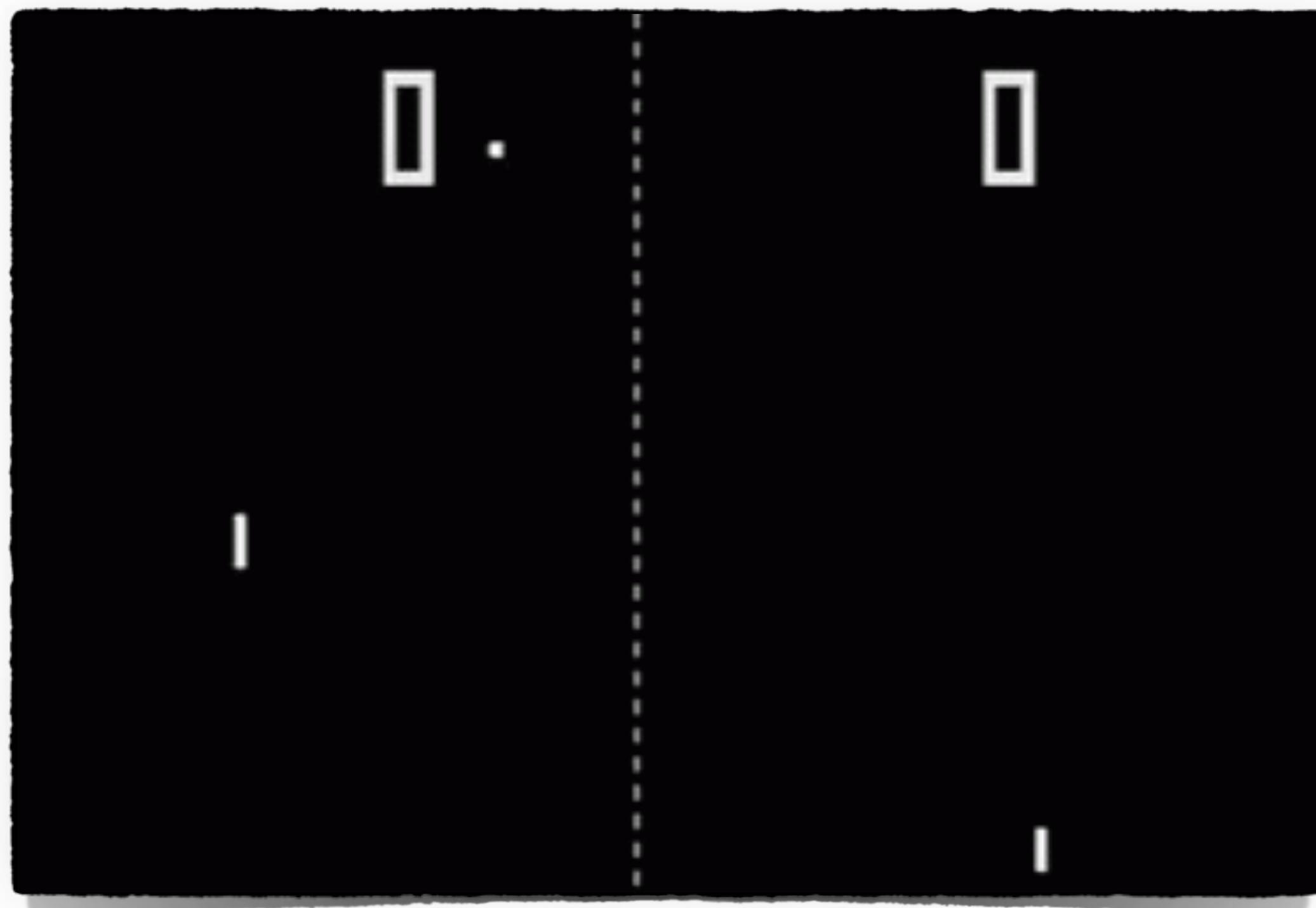
Como se  
desenvol

**Desenvolvendo  
jogos!**

Nesse semestre, você irá implementar 4 jogos clássicos e desenvolver o seu próprio jogo!

- ▶ **Em C++!**
- ▶ **Sem nenhuma engine! (apenas com SDL)**

# 1. Pong



Atari, 1972 (Arcade)

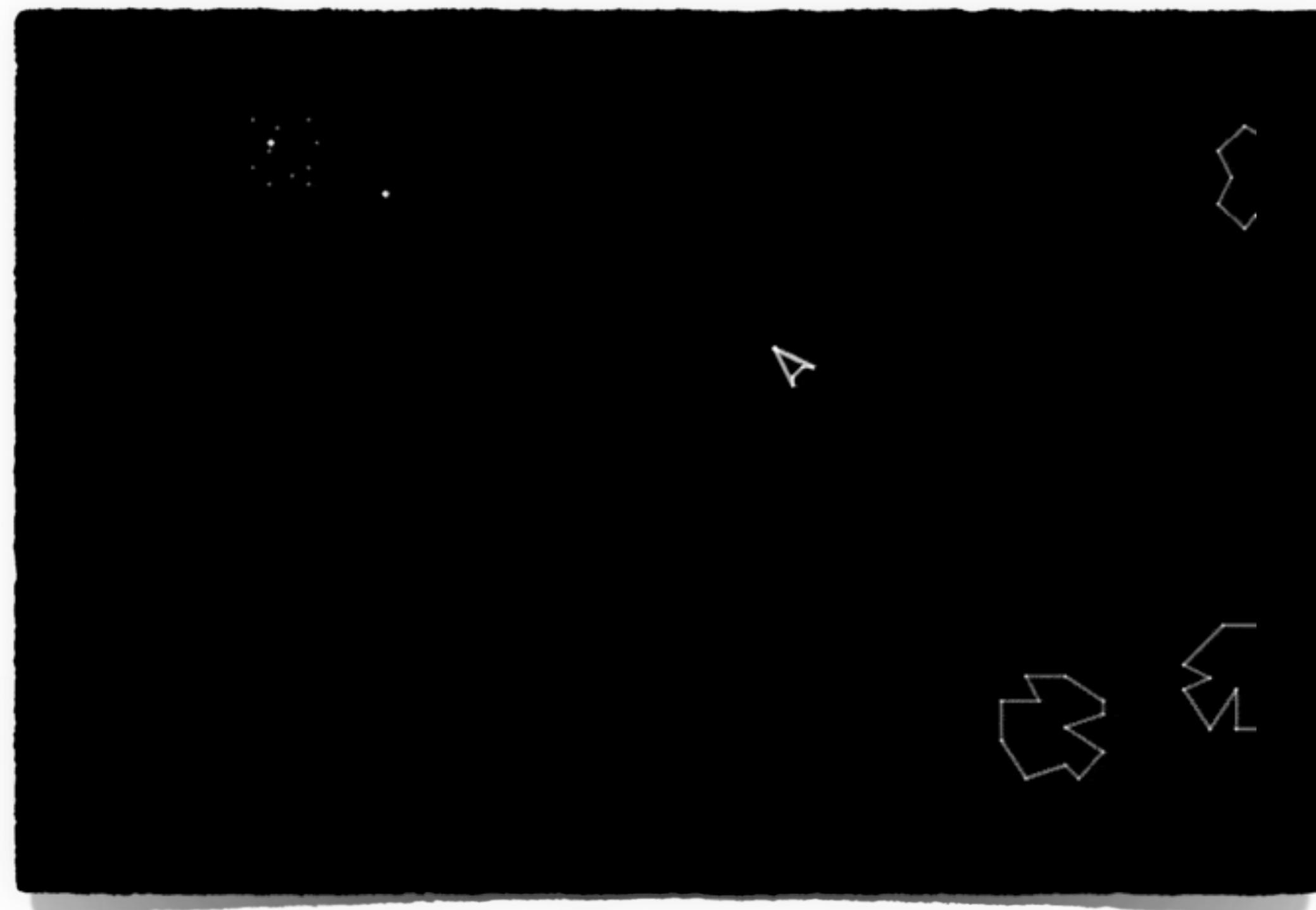
## Game Loop

- ▶ Entrada, atualização e saída
- ▶ Gerenciamento de quadros

## Modelagem de objetos

- ▶ Hierarquia de classes
- ▶ Componentes

## 2. Asteroids



Atari, 1979 (Arcade)

### Física

- ▶ Forças e objetos rígidos
- ▶ Detecção de colisão
- ▶ Resolução de colisão

### Gráficos 2D

- ▶ Gráficos Vetoriais
- ▶ Sistemas de Partículas

### 3. Super Mario Bros



Nintendo, 1983 (NES)

#### Física

- ▶ Resolução de colisões

#### Gráficos 2D

- ▶ Sprites e spritesheets
- ▶ Animações
- ▶ Tilemaps

## 4. Star Fox Tunel

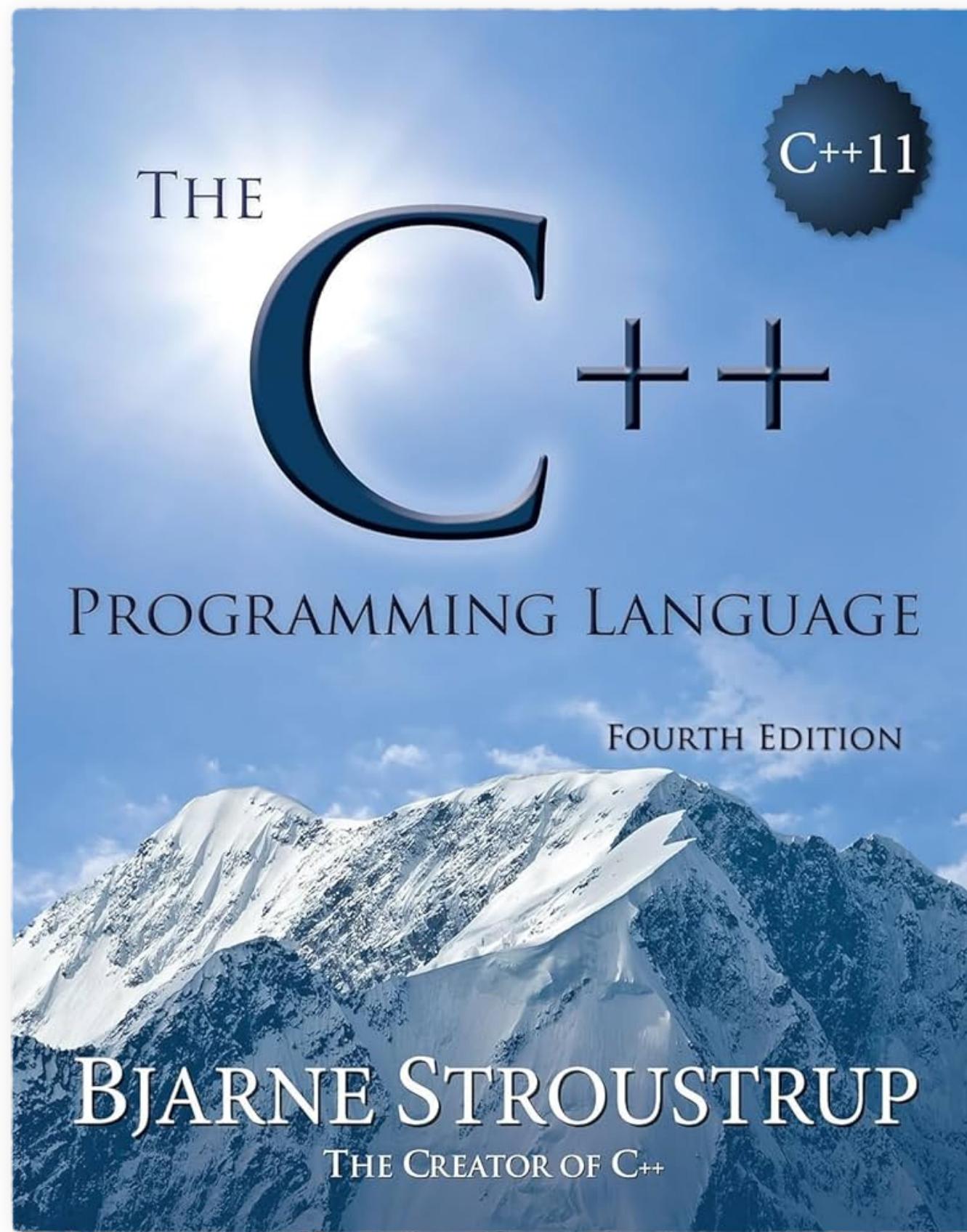


### Gráficos 3D

- ▶ Pipeline gráfico
- ▶ Modelos 3D
- ▶ Câmeras
- ▶ Iluminação e sombreamento
- ▶ Visibilidade

Nintendo, 1992 (N64)

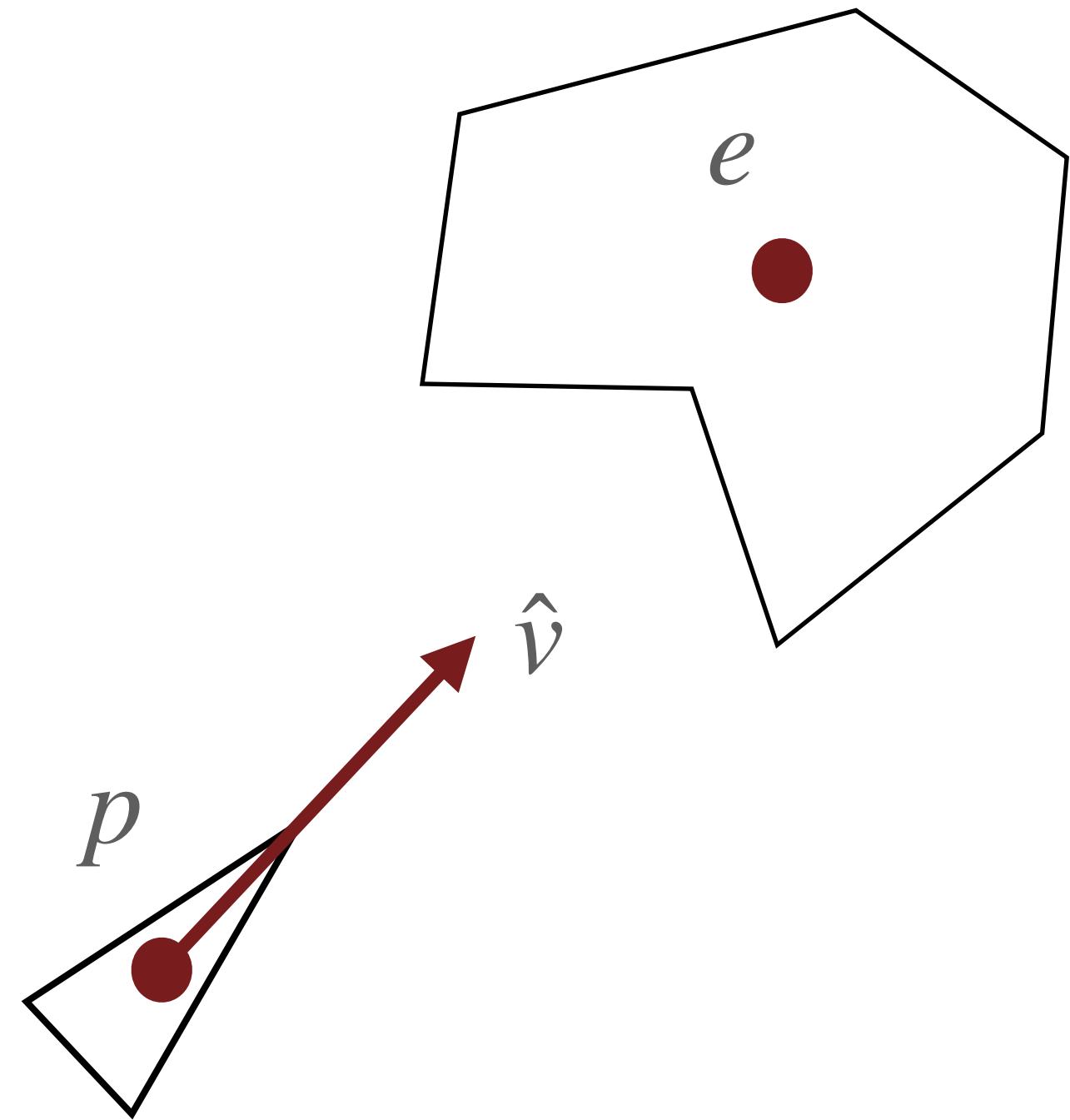
# Por que C++?



- ▶ A maioria dos motores, até mesmo o Unity, são escritos em C++
- ▶ A maioria dos desenvolvedores AAA
  - ▶ Riot, Blizzard, Naughty Dog, ...
- ▶ Ainda buscam principalmente experiência em C++
- ▶ Mais experiência em C++ → melhor currículo

# Objetivo

*Essa disciplina é uma introdução às técnicas fundamentais para a programação de jogos 2D (e 3D), com o objetivo de possibilitar que os alunos criem um portfólio inicial para uma carreira na indústria de jogos.*



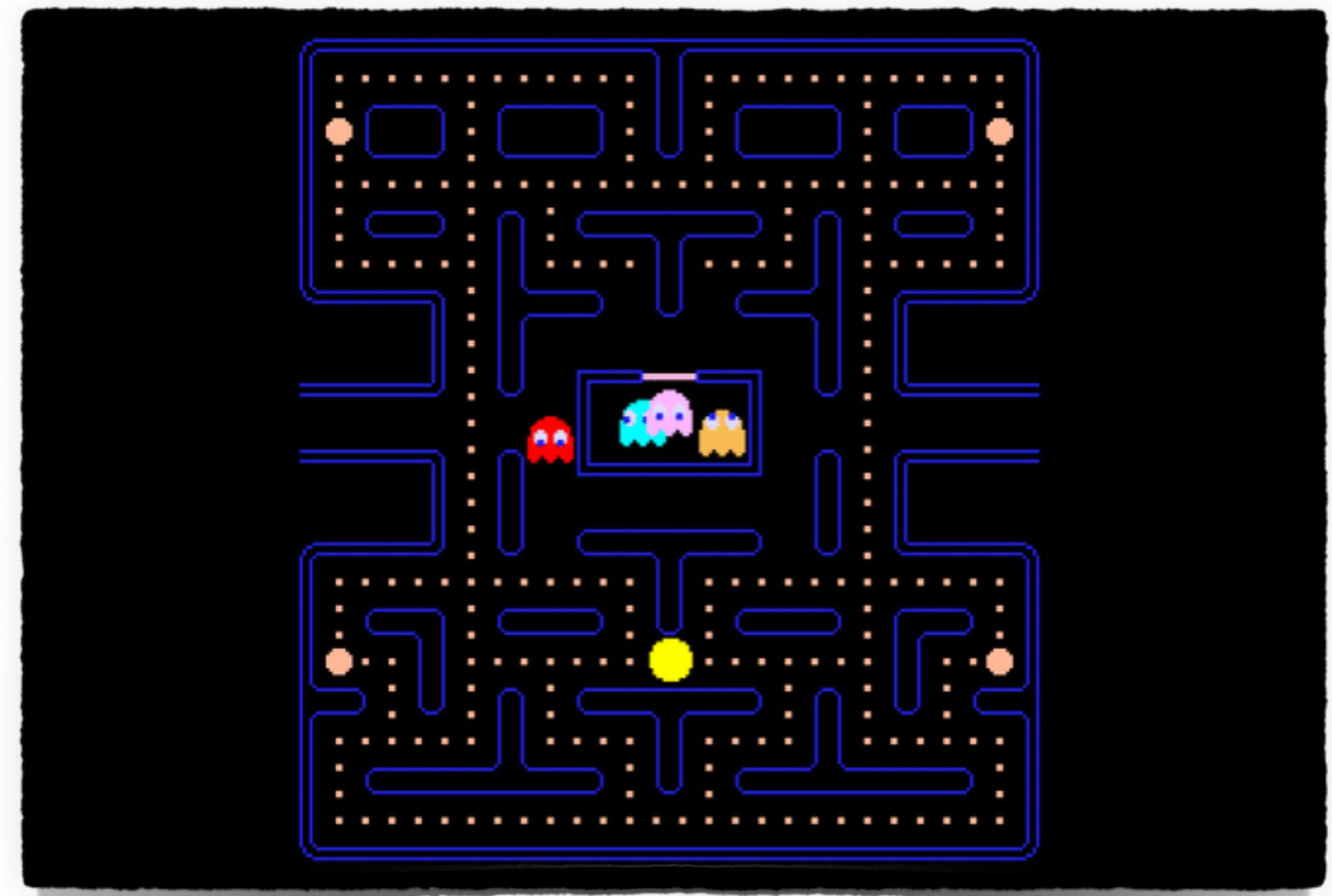
## Matemática para Jogos

- ▶ Operações básicas de vetores e matrizes
- ▶ Sistemas de coordenadas
- ▶ Transformações geométricas

# Além disso

## Inteligência Artificial

- ▶ Máquina de estados finitos
- ▶ Pathfinding
- ▶ Game Playing
- ▶ Procedural Content Generation



IA dos fantasmas do PacMan

# Além disso

## Áudio

- ▶ Processamento digital de áudio
- ▶ Sintetizadores
- ▶ Efeitos e mixagem
- ▶ Áudio 3D e Adaptativo
- ▶ Middlewares de áudio



Chip de áudio do SNES

# Pré-requisitos

## DCC205: Estrutura de dados

- ▶ Programação em C++
- ▶ Estrutura de dados fundamentais e algoritmos associados

## MAT038: Geometria Analítica e Álgebra Linear

- ▶ Operações básicas com vetores e matrizes

# Avaliação

- ▶ Trabalhos Práticos (40%)
- ▶ Projeto Final (60%)
  - ▶ *Game Design Document* (10%)
  - ▶ Prototipação (20%)
  - ▶ Playtest (10%)
  - ▶ Produto Final (20%)

# Trabalhos práticos

Implementação **individual** de mecânicas de jogos clássicos em C++, com duração de 1-2 semanas.

- ▶ TP0: Configuração Inicial
- ▶ TP1: Pong
- ▶ TP2: Asteroids
- ▶ TP3: Super Mario Bros (1-1)
- ▶ TP4: Star Fox Tunel

**Entregas via Moodle (link para o repo GitHub)**

# Projeto final

Proposta, implementação e teste de um pequeno jogo completo, **em grupos** (2-4 alunos), com duração de aproximadamente 7.5 semanas.

- ▶ PF1: Documento de Design (1 semana)
- ▶ PF2: Protótipo (4 semanas)
- ▶ PF3: Playtest (1 semana)
- ▶ PF4: Finalização (1.5 semanas)

**Entregas via Moodle ([link para o repo GitHub](#))**

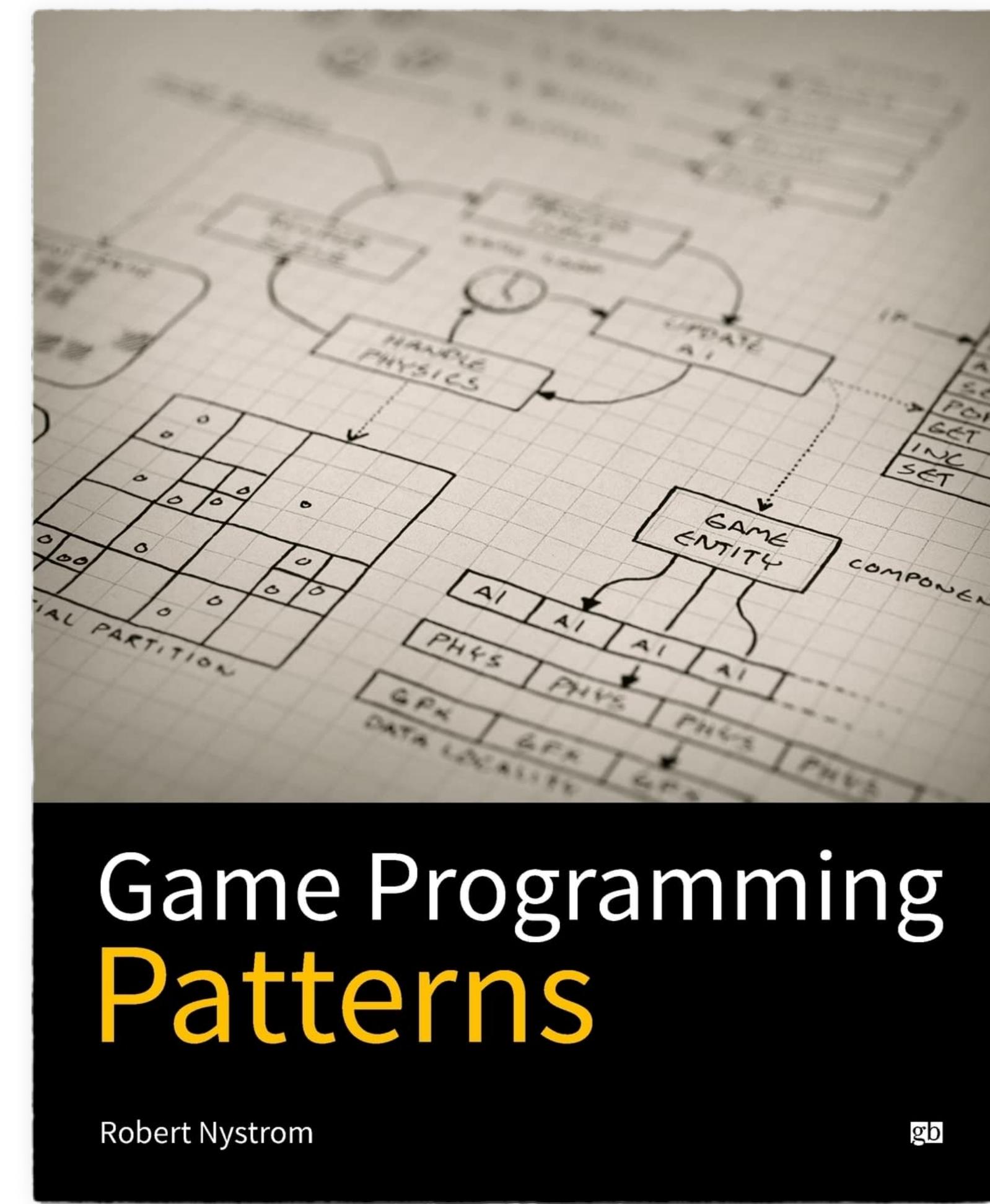
# Calendário

Semana	Data	Aula	Leituras	Trabalhos Práticos (TPs)	Projeto Final
1	10/03	Introdução			
	12/03	SDL			
2	17/03	Eng Soft: Game Loop		TP0: Setup	
	19/03	Eng Soft: Game Objects			
3	24/03	Lab 1: Pong			
	26/03	Física: Vetores e Movimentação			
4	31/03	Física: Detecção de Colisão			
	02/04	Gráficos 2D: Spritesheets e Tilemap			
5	07/04	Lab 2: Asteroids			
	09/04	Gráficos 2D: Câmeras, Menus e HUD			
6	14/04	Gráficos 2D: Sistema de Partículas		TP2: Asteroids	
	16/04	Game Design			
7	21/04	<b>Feriado (Tiradentes)</b>			
	23/04	Física: Resolução de Colisão			Game Design Document
8	28/04	Lab 3: Super Mario Bros			
	30/04	Gráficos 3D: Pipeline Gráfico		TP3: Super Mario Bros	
9	05/05	Gráficos 3D: Objetos 3D			
	07/05	Gráficos 3D: Transformações			
10	12/05	Lab 4: Star Fox		TP4: Star Fox Tunel	
	14/05	Gráficos 3D: Cores e Texturas			
11	19/05	Gráficos 3D: Shaders		Prototipação	
	21/05	Gráficos 3D: Visibilidade			
12	26/05	IA: Máquinas de Estados Finitos			
	28/05	IA: Pathfinding			
13	02/06	IA: Game Playing			
	04/06	IA: Procedural Content Generation			
14	09/06	Áudio: representações		Playtest	
	11/06	Áudio: síntese			
15	16/06	Áudio: sequenciamento			
	18/06	Eng Soft: Debug & Teste			
16	23/06	Eng Soft: Playtesting		Finalização	
	25/06	Tópicos Avançados: Game Engines			
17	30/06	Tópicos Avançados: Jogos Multiplayer			
	02/07	Conclusão			
18	07/07	Apresentação dos Trabalhos Finais I			
	09/07	Apresentação dos Trabalhos Finais II			

[https://docs.google.com/spreadsheets/d/1\\_NvioUqL5Fw3aMp-xrZKNVDo0eUkiw9YFch0rEERhg/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1_NvioUqL5Fw3aMp-xrZKNVDo0eUkiw9YFch0rEERhg/edit?usp=sharing)

# Materiais – Leituras

- ▶ *Game Programming Patterns*, Robert Nystrom
- ▶ *Beginning Game Programming v2.0*, Lazy Foo
- ▶ *Game Programming in C++*, Sanjay Madhav
- ▶ *The Nature of Code*, Daniel Shiffman



# Materiais – Programação

m

## ▶ Simple DirectMedia Layer (SDL)

*Biblioteca C/C++ para facilitar o acesso aos dispositivos multimedia (video, áudio, rede, entrada, etc) em diferentes plataformas*

## ▶ IDE CLion e Github

*Utilizaremos a IDE CLion e o Github durante todos os projetos práticos, para simular um ambiente real de desenvolvimento de jogos*



# Comunicação

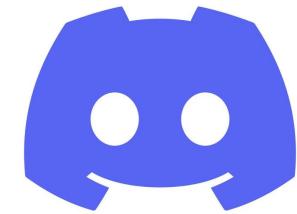


## Email

- ▶ Questões referentes a notas e frequência (~2 dias de resposta)

## Discord

- ▶ Questões referentes ao conteúdo da disciplina (~30 minutos de resposta)



<https://discord.gg/fcUZsSE2>

## Atendimento

- ▶ Agendar por email ou Discord

## Atrasos

- ▶ Penalização de 15% para cada dia de atraso
- ▶ Cada atraso pode ser de no máximo 2 dias

## Compartilhamento

- ▶ É permitido:
  - ▶ Trabalhar com um colega para solucionar um trabalho prático (notificar no cabeçalho do seu código)
  - ▶ Tirar dúvidas no Discord sobre como resolver um problema que está tendo no seu código
- ▶ Não é permitido:
  - ▶ Compartilhar a solução dos seus trabalhos práticos (parcial ou na íntegra) com seus colegas
  - ▶ Postar a solução dos seus trabalhos práticos (parcial ou na íntegra) publicamente na Internet (e.g., GitHub)

# Site da disciplina

O moodle será utilizado apenas para reportar notas e feedback.

Todas as informações e avisos podem ser encontrados na página da disciplina:

**<https://lucasnfe.github.io/dcc192-2025-1>**

UFMG - DCC192

Search UFMG - DCC192

Lucas N. Ferreira    DCC    Universidade Federal de Minas Gerais

Avaliações  
Avisos  
Calendário  
Materiais  
Sobre

DCC192 - Desenvolvimento de Jogos Digitais (2025/1)  
Essa disciplina é uma introdução às técnicas fundamentais para a programação de jogos 2D e 3D. Os alunos são apresentados a conceitos de projeto de software, física, gráficos, inteligência artificial e áudio aplicados para o desenvolvimento de jogos. Além disso, eles utilizam ferramentas profissionais para simular um ambiente de desenvolvimento real (como um estúdio de jogos), tendo a oportunidade de publicar um portfólio pessoal com os trabalhos desenvolvidos ao longo do curso.

Avisos

Semana 1  
Mar 11 · 0 min read

- Seja bem-vindo(a) à DCC192 - Desenvolvimento de Jogos Digitais!

Aulas

- Segundas e Quartas, 19:00-20:40pm, ICEX1017

Professor

 Lucas N. Ferreira  
lferreira@dcc.ufmg.br

Sala DCC4311

This site uses [Just the Docs](#), a documentation theme for Jekyll.

# Próxima aula

## A2: Configuração Inicial

- ▶ Instalar e configurar a SDL na CLion
- ▶ Escrever seu primeiro programa em SDL
- ▶ Janelas
- ▶ Eventos de Entrada
- ▶ Geometrias básicas