

## Estudo da Computabilidade e Complexidade de Algoritmos no Desenvolvimento de um Jogo Digital: Aplicações no Pacman

*Study of Computability and Algorithmic Complexity in the Development of a Digital Game: Applications in Pacman*

Vinícius Olivar Hammes Rodegheri<sup>1</sup>, Kadidja Valeria Reginaldo de Oliveira

<sup>1</sup> Centro Universitário do Distrito Federal UDF, Coordenação de Tecnologia da Informação, Brasília, DF

<sup>2</sup> Centro Universitário do Distrito Federal UDF, Coordenação de Tecnologia da Informação, Brasília, DF

### RESUMO:

**Introdução:** Este trabalho investiga a aplicação de conceitos de computabilidade e complexidade de algoritmos no desenvolvimento de um jogo digital baseado no Pacman. O objetivo principal é analisar como os algoritmos de movimentação do personagem, lógica de colisão e inteligência artificial (IA) dos fantasmas impactam o desempenho do jogo. Através de uma análise das escolhas algorítmicas, discutem-se as complexidades computacionais envolvidas, com ênfase na otimização de recursos e na melhoria da experiência do jogador. O estudo também explora a relação entre os algoritmos implementados e a jogabilidade, propondo aprimoramentos, como a implementação de algoritmos de busca mais eficientes e técnicas de otimização de memória. Ao final, o trabalho demonstra como os conceitos de computabilidade podem ser aplicados na prática para criar jogos digitais mais eficientes e desafiadores.

**Palavras-chave:** Computabilidade, Algoritmos, Complexidade, Jogo Digital, Pacman.

**Palavras-chave:** Computabilidade, Algoritmos, Complexidade, Jogo Digital, Pacman, Python.

### ABSTRACT:

**Introduction:** This work investigates the application of concepts of computability and algorithmic complexity in the development of a digital game based on Pacman. The main objective is to analyze how the movement algorithms of the character, collision logic, and the artificial intelligence (AI) of the ghosts impact the game's performance. Through an analysis of the algorithmic choices, the computational complexities involved are discussed, with an emphasis on resource optimization and improving the player experience. The study also explores the relationship between the implemented algorithms and gameplay, proposing improvements such as the implementation of more efficient search algorithms and memory optimization techniques. In the end, the work demonstrates how concepts of computability can be applied in practice to create more efficient and challenging digital games.

**Keywords:** Computability, Algorithms, Complexity, Digital Game, Pacman, Python.

## INTRODUÇÃO

A evolução dos jogos digitais é um reflexo do constante avanço da tecnologia computacional, especialmente no que diz respeito à aplicação de conceitos de computabilidade e eficiência algorítmica. Neste contexto, o jogo Pacman foi escolhido como objeto de estudo para explorar como as técnicas de desenvolvimento podem ser aplicadas de forma eficiente e escalável.

A disciplina de Computabilidade e Complexidade de Algoritmos aborda questões cruciais no desenvolvimento de sistemas computacionais, incluindo a análise de algoritmos e a identificação de problemas de diferentes classes de complexidade, como P, NP e NP-Completo. Este trabalho objetiva investigar como essas noções podem ser implementadas de maneira prática no desenvolvimento de um jogo digital. A partir da construção de um modelo simples do Pacman, serão analisados os algoritmos de movimentação, de tomada de decisão da IA dos fantasmas, e as operações de manipulação de imagens.

A principal contribuição deste estudo é demonstrar como técnicas de análise de complexidade e computabilidade podem ser utilizadas para melhorar a experiência do usuário sem comprometer o desempenho do sistema. Este trabalho também destaca o processo de otimização de algoritmos, propondo melhorias para tornar o jogo mais desafiador e eficiente.

## METODOLOGIA

A construção do jogo Pacman seguiu um processo iterativo, dividido em etapas claras que permitiram o desenvolvimento e a análise dos algoritmos propostos. O jogo foi projetado para ser simples, com foco na movimentação do personagem, nas interações com os fantasmas e na coleta de itens.

### 1. Planejamento e Arquitetura do Sistema

O projeto foi estruturado de maneira modular, com cada componente do jogo desenvolvido separadamente para facilitar a manutenção e os testes. Os módulos principais incluem:

- **Movimentação do Pacman:** A movimentação foi programada para ser responsiva, com base nas teclas pressionadas pelo jogador, alterando as coordenadas do Pacman na matriz que representa o labirinto.

- **IA dos Fantasmas:** A lógica de movimentação dos fantasmas foi inicialmente implementada com uma abordagem simples e gulosa, com o objetivo de perseguir o Pacman de maneira previsível.
- **Manipulação de Imagens:** Utilizando as bibliotecas OpenCV e Pygame, o jogo foi projetado para renderizar sprites e realizar as transformações gráficas de forma eficiente.
- **Sistema de Colisão:** A verificação de colisão foi implementada para identificar interações entre o Pacman, os fantasmas e os itens do labirinto.

## 2. Técnicas e Algoritmos Empregados

A implementação do jogo foi guiada por conceitos de eficiência algorítmica:

- **Divisão e Conquista:** O mapa do jogo foi dividido em blocos, e cada bloco foi tratado individualmente para otimizar a busca por colisões.
- **Algoritmos Gulosos:** A IA dos fantasmas seguiu um algoritmo simples, tomando decisões locais baseadas apenas nas posições atuais, o que oferece uma boa base para a criação de comportamentos mais complexos futuramente.
- **Estruturas de Dados:** Utilizou-se principalmente matrizes e listas para armazenar as informações do jogo, representando o labirinto e os objetos do jogo de forma eficiente.

## 3. Ferramentas e Tecnologias

- **Linguagem Python** foi a principal utilizada para o desenvolvimento do código, devido à sua simplicidade e versatilidade.
- **Bibliotecas Utilizadas:**
  - **Pygame:** Para a renderização gráfica e controle de eventos de entrada do usuário.
  - **NumPy:** Para manipulação de arrays e otimização de cálculos relacionados às coordenadas do Pacman e das colisões.
  - **OpenCV:** Para manipulação e transformação de imagens no jogo, como escalonamento e rotação.

## 4. Validação e Testes

O jogo foi testado para garantir o correto funcionamento dos algoritmos:

- **Testes de Funcionalidade:** Verificações de que os movimentos, colisões e interações estavam funcionando conforme esperado.
- **Testes de Desempenho:** Avaliação do tempo de resposta do jogo e uso de memória em diferentes cenários de jogo.

## RESULTADOS

O jogo implementado apresentou bons resultados em termos de funcionalidade e desempenho. A movimentação do Pacman e a lógica de colisão foram eficientes, com um tempo de resposta adequado para a interação do jogador. A IA dos fantasmas, embora simples, foi eficaz em criar uma dinâmica de perseguição e evasão. No entanto, a análise revelou áreas que podem ser otimizadas, especialmente no que diz respeito à complexidade dos algoritmos de movimentação dos fantasmas e ao uso de memória no armazenamento do estado do jogo.

### Desempenho e Complexidade

- **Movimentação do Pacman:** A complexidade foi  $O(1)$  por movimento, devido à representação direta das coordenadas na matriz.
- **Movimentação dos Fantasmas:** A abordagem gulosa tem complexidade  $O(n)$ , onde  $n$  é o número de fantasmas no jogo, uma vez que cada fantasma toma uma decisão baseada no Pacman.
- **Sistema de Colisão:** A verificação de colisão teve complexidade  $O(1)$ , realizada diretamente sobre a matriz de dados do labirinto.

### Métricas de Desempenho

- **Tempo médio por ciclo de atualização:** 10 ms.
- **Uso de memória:** Aproximadamente 5 KB para um labirinto de 20x20.

## DISCUSSÃO

A análise da implementação do jogo Pacman revelou que os algoritmos empregados são eficientes para o escopo atual do jogo, mas que melhorias podem ser aplicadas, especialmente em termos de IA e otimização do uso de memória. A IA dos fantasmas, baseada em uma abordagem gulosa, pode ser aprimorada utilizando algoritmos de busca mais avançados, como  $A^*$ , para tornar o comportamento mais dinâmico e imprevisível.

Além disso, o sistema de colisão, embora eficiente para labirintos de tamanho médio, pode ser otimizado para jogos de maior escala. A aplicação de técnicas como compressão de dados ou o uso de estruturas de dados mais avançadas pode reduzir o consumo de



memória sem afetar a performance.

## CONCLUSÃO

Este trabalho demonstrou como os conceitos de computabilidade e complexidade algorítmica podem ser aplicados ao desenvolvimento de jogos digitais. A implementação do jogo Pacman não só resultou em um produto funcional e eficiente, mas também permitiu uma análise prática de como as escolhas algorítmicas impactam o desempenho do sistema. Futuras melhorias podem incluir a introdução de algoritmos de IA mais sofisticados, técnicas de otimização para o sistema de colisão e compressão de dados para maior escalabilidade.

## REFERÊNCIAS BIBLIOGRÁFICAS

ANTON, Howard; RORRES, Chris. **Álgebra Linear para Engenharia: Transformações de Fourier**. 9 ed. Cidade: LTC - Livros Técnicos e Científicos, 2019.

ASTH, Rafael. **Matrizes**. Toda Matéria, [s.d.]. Disponível em: <https://www.todamateria.com.br/matrizes-resumo/>. Acesso em: 20 mar. 2024.

MCKINNEY, Wes. **Python para análise de dados: Tratamento de dados com Pandas, NumPy e IPython**, Novatec Editora, 2018. Disponível em: [https://scholar.google.com.br/scholar?hl=ptBR&as\\_sdt=0,5&q=numpy+para+processamento+grafico#d=gs\\_qabs&t=1712784886339&u=%23p%3DEq587W1eKVAJ](https://scholar.google.com.br/scholar?hl=ptBR&as_sdt=0,5&q=numpy+para+processamento+grafico#d=gs_qabs&t=1712784886339&u=%23p%3DEq587W1eKVAJ). Acesso em: 3 abr. 2024.

CARNEIRO, Álvaro. **Técnicas de processamento digital de imagens no Python com matemática**, 2021. Disponível em: <https://medium.com/data-hackers/t%C3%A9cnicas-de-processamento-digital-de-imagens-no-python-com-matem%C3%A1tica-154eae204447>. Acesso em: 3 abr. 2024.

SILVA, Domiciano. **Translação de espelhos**, [s.d.]. Disponível em: <https://mundoeducacao.uol.com.br/fisica/translacao-espelhos.htm>. Acesso em: 20 mar. 2024.

A biblioteca **Matplotlib** (HUNTER, 2007), foi utilizada para gerar gráficos que permitiram a criação do personagem Pac-man a partir de um conjunto de dados. Disponível em: <https://matplotlib.org>. Acesso em: 3 abr. 2024.

A biblioteca **NumPy** (VAN DER WALT et al., 2011), foi utilizada para análise de

dados do personagem Pac-man. VAN DER WALT, S., COLBERT, S. C., & VAROQUAUX, G. (2011). The NumPy array: a structure for efficient numerical computation. Computing in Science & Engineering, 13(2), 22-30. Disponível em: <https://numpy.org>. Acesso em: 20 abr. 2024.

**A biblioteca OpenCV** (BRADSKI, 2000), foi utilizada para detecção de bordas no personagem Pac-man. BRADSKI, G. (2000). The OpenCV Library. Disponível em: <https://opencv.org>. Acesso em: 20 abr. 2024.

**A biblioteca Pandas** (MCKINNEY, 2010), foi utilizada para análise de dados do personagem Pac-man. MCKINNEY, W. (2010). Data structures for statistical computing in Python (Versão 1.3.3). Proceedings of the 9th Python in Science Conference, 445, 51-56. Disponível em: <https://pandas.pydata.org/>. Acesso em: 15 abr. 2024.