

# **INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO**

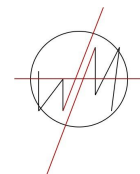
**CJOPROO - PROGRAMAÇÃO ORIENTADA A OBJETOS**

Professor: Paulo

## **Desenvolvimento de um Jogo de Ping Pong com a Biblioteca raylib em C++**

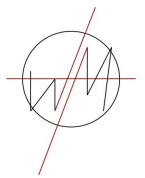
Wedyner Rodrigo Maciel – CJ3019462

Campos do Jordão  
2025



## Sumário

Desenvolvimento de um Jogo de Ping Pong com a Biblioteca raylib em C++.....	1
Sumário.....	2
<b>Resumo.....</b>	<b>3</b>
<b>1. Introdução.....</b>	<b>3</b>
<b>2. Metodologia.....</b>	<b>5</b>
<b>3. Resultados Obtidos.....</b>	<b>6</b>
<b>4. Conclusão.....</b>	<b>8</b>
<b>5. Referências Bibliográficas.....</b>	<b>10</b>



## Resumo

O presente trabalho descreve o processo de desenvolvimento de um jogo digital do tipo "Ping Pong", elaborado com a linguagem de programação C++ e utilizando a biblioteca gráfica raylib, voltada à criação de aplicações multimídia e jogos 2D. O projeto foi idealizado com o intuito de explorar, de forma prática e didática, conceitos fundamentais da programação orientada a objetos, bem como proporcionar uma introdução acessível ao desenvolvimento de jogos com manipulação de gráficos em tempo real.

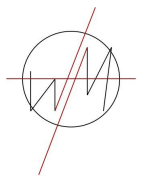
A estruturação do jogo foi baseada em classes específicas que representam os principais componentes da dinâmica do jogo: as raquetes, controladas pelos jogadores, e a bola, elemento central da interação. Cada classe foi projetada com métodos responsáveis por desenhar os objetos na tela, movimentá-los de acordo com as entradas do teclado e aplicar as regras de colisão e pontuação.

Durante o processo de desenvolvimento, foram implementadas rotinas de controle de colisão entre a bola e as raquetes, além da verificação de limites de tela, garantindo que a movimentação dos elementos fosse consistente com as regras do jogo. Também foi incorporado um sistema de pontuação simples, que reinicia a posição da bola ao final de cada jogada, mantendo o jogo dinâmico e contínuo.

Os testes realizados comprovaram a estabilidade da aplicação e a responsividade dos controles, permitindo uma experiência fluida e interativa para dois jogadores. A simplicidade da interface não comprometeu a funcionalidade, demonstrando que jogos eficazes podem ser desenvolvidos com poucos recursos gráficos, desde que a lógica e a estrutura do código estejam bem definidas.

Dessa forma, o projeto se mostrou eficaz tanto como uma ferramenta didática, aplicável em disciplinas de lógica de programação e computação gráfica, quanto como base para o desenvolvimento de jogos mais complexos e completos. A modularidade do código e a utilização de uma biblioteca de fácil manuseio como a raylib possibilitam futuras expansões, como a introdução de sons, interfaces gráficas mais ricas ou modos de jogo adicionais. O resultado final evidencia que o uso de tecnologias acessíveis pode viabilizar a criação de experiências digitais envolventes e educacionais.

## 1. Introdução



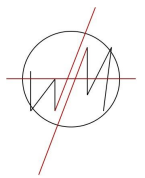
O desenvolvimento de jogos digitais tem se consolidado como uma das áreas mais dinâmicas e atrativas da ciência da computação, despertando o interesse de estudantes, profissionais e pesquisadores por envolver uma combinação rica de criatividade, lógica e domínio técnico. Além do entretenimento, os jogos oferecem um ambiente ideal para o aprendizado prático de diversos conceitos fundamentais da computação, como estruturas de dados, lógica de programação, algoritmos, orientação a objetos e manipulação gráfica. Ao criar jogos, os desenvolvedores são desafiados a pensar em tempo real, reagir a eventos do usuário, tratar colisões e aplicar estratégias para otimizar desempenho e tornar a experiência do usuário fluida e envolvente.

Neste contexto, este trabalho tem como objetivo principal o desenvolvimento de uma versão digital do clássico jogo "Ping Pong", também conhecido como "Pong", utilizando a linguagem de programação C++ e a biblioteca gráfica raylib. A escolha do jogo baseia-se em sua simplicidade estrutural e valor histórico, pois trata-se de um dos primeiros jogos eletrônicos da história, sendo frequentemente utilizado como ponto de partida no ensino de desenvolvimento de jogos. Sua lógica é direta e intuitiva, mas ao mesmo tempo proporciona um rico campo de aplicação de técnicas fundamentais da programação.

A linguagem C++ foi selecionada por sua ampla utilização no mercado de desenvolvimento de sistemas de alto desempenho, incluindo engines gráficas e jogos comerciais, além de sua forte orientação a objetos, que permite uma estruturação mais robusta e reutilizável do código. Por sua vez, a biblioteca raylib destaca-se por sua facilidade de uso, código aberto e foco na simplicidade, sendo especialmente recomendada para iniciantes e para projetos educacionais que envolvem programação gráfica. Com uma API clara e bem documentada, raylib permite a manipulação de gráficos 2D, entrada de usuário, controle de áudio e outros recursos essenciais para a construção de um jogo completo.

A implementação do projeto foi concebida com fins didáticos, oferecendo uma aplicação funcional que pode ser utilizada como material complementar em disciplinas como Lógica de Programação, Estrutura de Dados, Programação Orientada a Objetos e Computação Gráfica. O desenvolvimento do jogo também proporciona uma oportunidade concreta para os alunos compreenderem o ciclo completo de construção de uma aplicação interativa, desde a modelagem dos elementos até a manipulação dos eventos em tempo real.

Além disso, este tipo de projeto serve como estímulo para a criatividade dos estudantes, promovendo o pensamento crítico e o raciocínio lógico ao lidar com problemas



de programação em um contexto lúdico. A prática de desenvolver jogos simples como o Ping Pong prepara o terreno para desafios mais complexos, sendo um primeiro passo relevante rumo ao domínio de técnicas mais avançadas em desenvolvimento de software.

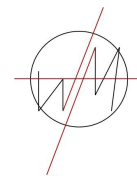
## 2. Metodologia

A metodologia adotada para o desenvolvimento do jogo "Ping Pong" seguiu uma abordagem incremental e estruturada, composta por etapas bem definidas, desde a escolha das ferramentas até a implementação e os testes finais da aplicação. Essa organização permitiu acompanhar com precisão a evolução do projeto, mantendo a modularidade do código, a clareza nas implementações e a facilidade de manutenção.

O primeiro passo consistiu na definição das ferramentas e tecnologias que seriam utilizadas ao longo do projeto. Optou-se pela linguagem de programação C++, amplamente reconhecida por seu desempenho, suporte robusto à programação orientada a objetos e presença consolidada na indústria de desenvolvimento de jogos. Para a construção da interface gráfica e controle dos elementos visuais, foi escolhida a biblioteca raylib, uma API leve e de fácil aprendizado, ideal para projetos educacionais e protótipos interativos em 2D.

O compilador g++ (GCC) foi utilizado para compilar o código-fonte, sendo uma escolha confiável, compatível com os padrões da linguagem e com ampla documentação disponível. O diferencial neste projeto foi o uso do Notepad++ como ambiente de desenvolvimento, em conjunto com a configuração manual do terminal do sistema operacional para compilar e executar os arquivos. O Notepad++ foi escolhido por ser um editor de texto leve, eficiente e compatível com realce de sintaxe para C++, além de permitir uma edição rápida e organizada do código-fonte. Embora não ofereça recursos avançados como depuração integrada, sua simplicidade foi suficiente para atender às demandas deste projeto, proporcionando um ambiente enxuto e funcional para o desenvolvimento com raylib.

A estruturação do código seguiu os princípios da programação orientada a objetos, com foco em clareza, modularização e reutilização. Foram definidas duas classes principais: **Paddle** e **Ball**. A classe **Paddle** é responsável pela representação das raquetes controladas pelos jogadores, incluindo atributos como posição e velocidade, bem como métodos para movimentação e renderização dos objetos na tela. A classe **Ball**, por sua vez, encapsula toda a lógica referente à bola do jogo, como deslocamento, detecção de colisões com as



bordas da tela e com as raquetes, reinício da posição após pontuações e controle de sua direção.

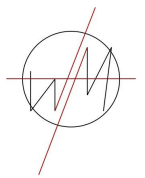
A lógica de funcionamento do jogo foi implementada de forma simples e direta, refletindo a dinâmica tradicional do clássico "Ping Pong". O controle das raquetes pelos jogadores é realizado através de teclas específicas: o Jogador 1 utiliza as teclas W e S para mover a raquete para cima e para baixo, respectivamente, enquanto o Jogador 2 utiliza as setas direcionais ↑ e ↓. A bola se desloca em linha reta com velocidade constante e altera sua trajetória ao colidir com as raquetes ou com os limites superior e inferior da janela. Quando a bola ultrapassa o limite esquerdo ou direito da tela, um ponto é atribuído ao jogador adversário e a bola retorna ao centro da tela com nova direção.

Foram realizados testes manuais com foco na verificação da estabilidade do sistema, funcionalidade das colisões e fluidez da jogabilidade. Esses testes abrangeram diferentes resoluções de tela e velocidades da bola, de forma a garantir que a experiência do jogo permanecesse consistente em diversos contextos de execução. A resposta aos comandos foi avaliada quanto à precisão e tempo de reação, e o sistema de pontuação foi validado em diferentes cenários de jogo. O resultado dos testes foi satisfatório, demonstrando que a aplicação possui um desempenho estável e responde adequadamente às interações do usuário.

Com isso, a metodologia empregada demonstrou-se eficaz para atingir os objetivos definidos, proporcionando uma implementação bem organizada, funcional e didática. A utilização de ferramentas acessíveis como o Notepad++ em conjunto com raylib provou ser uma solução viável para projetos de pequeno porte, especialmente no contexto educacional, permitindo que estudantes compreendam na prática os fundamentos do desenvolvimento de jogos em C++ com interfaces gráficas simples.

### 3. Resultados Obtidos

Os resultados obtidos com a implementação do jogo "Ping Pong" utilizando C++ e a biblioteca gráfica raylib demonstraram que o projeto atingiu seus objetivos principais de forma eficaz. Os testes realizados, de natureza manual, evidenciaram que a aplicação



apresenta desempenho satisfatório e estável em diversos contextos de execução. O jogo manteve uma taxa de atualização constante de 60 quadros por segundo (FPS), o que garantiu uma fluidez visual agradável, sem travamentos ou perda de desempenho perceptível durante as partidas.

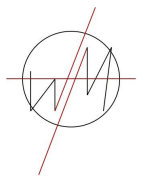
Um dos principais aspectos avaliados foi a responsividade dos controles. Observou-se que a movimentação das raquetes, controladas por diferentes pares de teclas para os dois jogadores, ocorre de maneira fluida, com resposta imediata aos comandos do teclado. Essa responsividade é essencial para garantir uma boa jogabilidade, especialmente em jogos que exigem reflexos rápidos e precisão nas ações, como é o caso de um jogo de Ping Pong.

A detecção de colisões entre a bola e os demais elementos do jogo — como as raquetes e os limites superior e inferior da janela — funcionou conforme o esperado. A bola altera sua direção de forma precisa ao entrar em contato com esses elementos, simulando corretamente as interações físicas básicas do jogo original. Essa funcionalidade é fundamental para manter a lógica do jogo coerente e proporcionar uma experiência que remeta ao comportamento clássico esperado pelos jogadores.

O sistema de pontuação também foi validado com sucesso. A cada vez que a bola ultrapassa a borda esquerda ou direita da tela, um ponto é corretamente atribuído ao jogador adversário. Imediatamente após o ponto, a bola é reposicionada no centro da tela e reiniciada com uma nova direção, permitindo a continuidade do jogo sem a necessidade de reinicialização manual ou intervenção externa. Essa funcionalidade automatizada proporciona uma dinâmica fluida e contínua durante a execução do jogo.

Embora a interface gráfica seja simples e minimalista, sua funcionalidade foi avaliada positivamente. Ela cumpre bem seu propósito, exibindo os elementos essenciais — como raquetes, bola e pontuação — de forma clara e objetiva. Além disso, o uso de uma estrutura orientada a objetos, com classes específicas para cada elemento do jogo, tornou o código-fonte organizado, modular e extensível. Isso permite que futuras melhorias possam ser facilmente implementadas, como a adição de efeitos sonoros, telas de menu, níveis de dificuldade, placar digital mais elaborado, ou até mesmo inteligência artificial para um modo de jogador único.

Esses resultados indicam que o jogo está apto não apenas para ser utilizado como ferramenta didática no ensino de programação e computação gráfica, mas também como base para projetos mais complexos no contexto do desenvolvimento de jogos. A



combinação entre desempenho, simplicidade e clareza no código mostra que a proposta de utilizar raylib e C++ em um ambiente acessível como o Notepad++ foi plenamente válida, especialmente em ambientes acadêmicos com foco na aprendizagem prática.

## 4. Conclusão

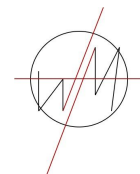
O desenvolvimento do jogo "Ping Pong" utilizando a linguagem C++ e a biblioteca gráfica raylib proporcionou uma experiência prática enriquecedora no contexto da programação orientada a objetos e do uso de bibliotecas gráficas 2D. A partir da estruturação do código em classes bem definidas, foi possível aplicar de forma concreta os princípios de encapsulamento, modularidade e reutilização, conceitos essenciais para a construção de aplicações robustas e escaláveis.

Ao longo do projeto, a raylib demonstrou ser uma ferramenta altamente eficiente para o ensino e a prototipação rápida de jogos, graças à sua interface simplificada e ao suporte multiplataforma. A combinação da raylib com um ambiente leve como o Notepad++, mesmo sem recursos avançados de desenvolvimento integrado, revelou-se viável e produtiva, especialmente em contextos educacionais e introdutórios. O uso dessa abordagem permitiu que o foco permanecesse no desenvolvimento da lógica do jogo e na estruturação do código, reduzindo as distrações e dificuldades relacionadas à configuração de ambientes complexos.

A aplicação final apresentou um desempenho sólido, com funcionamento fluido e resposta precisa aos comandos dos usuários. A estrutura implementada atendeu plenamente aos objetivos propostos, simulando de forma fiel a dinâmica do jogo de Ping Pong em sua versão clássica. A simplicidade da interface não comprometeu a experiência do usuário, e a estabilidade do sistema permitiu partidas contínuas sem falhas ou interrupções inesperadas. Além disso, o projeto se mostrou didaticamente eficiente, podendo ser utilizado como base em disciplinas de programação, computação gráfica ou desenvolvimento de jogos.

Outro ponto relevante observado durante o desenvolvimento foi a capacidade de extensão do projeto. A arquitetura orientada a objetos facilita significativamente a adição de novas funcionalidades, o que abre espaço para aprimoramentos e explorações futuras. Dentre as sugestões para trabalhos futuros, destaca-se a inclusão de um menu interativo,





permitindo ao jogador iniciar, pausar ou encerrar a partida de forma intuitiva. A adição de efeitos sonoros e trilha sonora também é uma possibilidade interessante para tornar o jogo mais imersivo e atrativo. Outro aprimoramento relevante seria a implementação de uma inteligência artificial simples, que permitiria ao jogador enfrentar o computador em um modo individual, aumentando a complexidade e o valor de replay do jogo.

Em suma, este projeto cumpriu seu papel de integrar teoria e prática no desenvolvimento de jogos digitais, oferecendo uma base sólida para aplicações mais sofisticadas e servindo como ferramenta de apoio ao ensino de conceitos fundamentais de programação. A clareza na estrutura, o desempenho estável e a possibilidade de evolução indicam que este trabalho não apenas atingiu seus objetivos iniciais, como também estabelece um ponto de partida promissor para novos desafios na área de desenvolvimento de jogos.

## 5. Referências Bibliográficas

ALVES, R. C.; MOURA, L. R. Desenvolvimento de jogos digitais com C++: conceitos e práticas. São Paulo: Novatec, 2019.

BURGES, David. *Beginning C++ Game Programming*. 2. ed. Birmingham: Packt Publishing, 2021.

DEITEL, Harvey M.; DEITEL, Paul J. *Como programar em C++*. 5. ed. São Paulo: Pearson Prentice Hall, 2005.

DEVMEDIA. *SFML, SDL e Raylib: comparação de bibliotecas para jogos*. Disponível em: <https://www.devmedia.com.br/sfml-sdl-e-raylib-comparacao-de-bibliotecas-para-jogos/40235>. Acesso em: 04 abr. 2025.

HABGOOD, J.; OVERMARS, M. *Fundamentos do desenvolvimento de jogos*. São Paulo: Cengage Learning, 2012.

KNUDSEN, M. *Mastering Raylib: A Complete Guide to Video Game Programming with Raylib*. 1. ed. Independently published, 2022.

RAYLIB. *Raylib: A simple and easy-to-use library to enjoy videogame programming*. Disponível em: <https://www.raylib.com>. Acesso em: 04 abr. 2025.

SANTOS, Lucas R.; OLIVEIRA, João P. *Introdução ao desenvolvimento de jogos em C++ com Raylib*. Revista Computação Aplicada, v. 15, n. 2, p. 35–48, 2022.

SILVA, F. A.; MENEZES, V. T. *Desenvolvimento de jogos com raylib: um estudo de caso em ambientes acadêmicos*. Anais do Simpósio Brasileiro de Games, 2023.

STRIEGLER, A. *C++ Fundamentals for Game Developers*. 3. ed. Boston: Apress, 2020.