

# Barbenheimer: Uma nova versão do clássico *Bomberman*

Gabrielly Batista de Freitas

Nathalia Macena da Silva

Adria Carolina de Melo Dias

Carlos Roberto da Silva Oliveira

Universidade de Brasília, Dept. de Ciência da Computação, Brasil

## Abstract

Bomberman (1983) é um jogo estratégico de grande sucesso, que acumula mais de 60 versões desde o seu lançamento. O presente trabalho objetiva detalhar o desenvolvimento de um jogo inspirado em Bomberman, utilizando a arquitetura RISC-V 32 bits. Durante a realização do projeto, foram encontrados muitos desafios na implementação de funcionalidades com o uso de uma linguagem de programação de baixo nível, porém, sendo realizada a implementação de um jogo, conforme proposto pelo projeto final, auxiliando na consolidação dos assuntos da disciplina.

**Keywords:** Bomberman; RISC-V; Assembly; RARS.

## 1 Introdução

Bomberman, criado originalmente em 1983, por Shinichi Nakamoto e Shigeki Fujiwara, e produzido pela desenvolvedora Hudson Soft, é uma franquia de jogos de estratégia, que possui mais de 60 jogos lançados até o momento. O jogo acompanha o herói “Bomberman”, também conhecido como “Shirobon”, um robô que, com o objetivo de escapar para a superfície e se tornar humano, utiliza bombas para derrotar seus inimigos e destruir paredes. A jogabilidade é simples, possuindo movimentação restrita a quatro direções (esquerda, direita, cima e baixo), e uma tecla adicional para depositar bombas, que geram uma explosão em formato de cruz após instantes, capaz de derrotar inimigos e abrir caminhos através de blocos destrutíveis. O objetivo das fases é encontrar a saída após eliminar todos os inimigos [Wikipédia ].

Com o objetivo de implementar um novo jogo baseado em Bomberman, contendo suas características funcionais principais similares ao jogo original, porém com uma temática diferente, o jogo desenvolvido neste projeto foi inspirado na rivalidade ocorrida durante o lançamento do filme *Barbie*, de Greta Gerwig, sobre a boneca americana em formato live-action, e do filme *Oppenheimer*, de Christopher Nolan, sobre a história do físico responsável pelo Projeto Manhattan. As duas obras, lançadas em 21 de julho de 2023, devido suas diferenças notórias em estética e temática, geraram grande repercussão por parte do público [Oldenzien 2024].

Dessa forma, o presente trabalho tem como objetivo detalhar o desenvolvimento de um jogo inspirado em Bomberman, utilizando Assembly RISC-V.



Figure 2: Pôsteres promocionais de Oppenheimer e Barbie

## 2 Metodologia

### 2.1 Ferramentas

Para o desenvolvimento do jogo, foi utilizado o RARS (RISC-V Assembler and Runtime Simulator), um simulador e montador de códigos em Assembly RISC-V 32 bits, baseado em Java [Sanderson ]. Para implementar a interface gráfica, foi utilizada a ferramenta “Bitmap Display”, do RARS, onde cada byte de memória em um endereço específico corresponde a um pixel exibido, além de conter dois frames (0 e 1), permitindo o uso de animações [de Paula ][Sanderson ]. Outra ferramenta do RARS utilizada foi o “Keyboard And Display MMIO Simulator”, que permite entradas a partir do pressionamento de teclas, e um dispositivo de saída [Sanderson ].

A realização das artes foi feita utilizando o programa Paint.NET e Adobe Photoshop. O Paint.NET é um editor gratuito voltado para criação de pixel art, enquanto o Adobe Photoshop é uma ferramenta profissional de edição de imagens que fornece recursos mais avançados enquanto a detalhes e precisão em pixel arts. As imagens foram convertidas para arquivo data a partir de um script feito por um ex-aluno, que permitiu a transformar arquivos bmp em data rapidamente. As artes do início do jogo, próxima fase e game over foram geradas pela IA ChatGPT. Embora tenham sido criadas por IA, as imagens passaram por ajustes manuais para melhorar sua qualidade e adequação ao tema do jogo.

Para obtenção da trilha sonora, foi utilizada a ferramenta de sintetizador MIDI do simulador RARS, que possibilita a reprodução de notas musicais a partir de códigos escritos em Assembly. As notas musicais foram extraídas de um site educacional Hooktheory, voltado ao ensino de teoria musical. Para converter essas notas em formato compatível com o simulador RARS, foi utilizado um código desenvolvido por um ex-aluno, que transforma as notas mu-

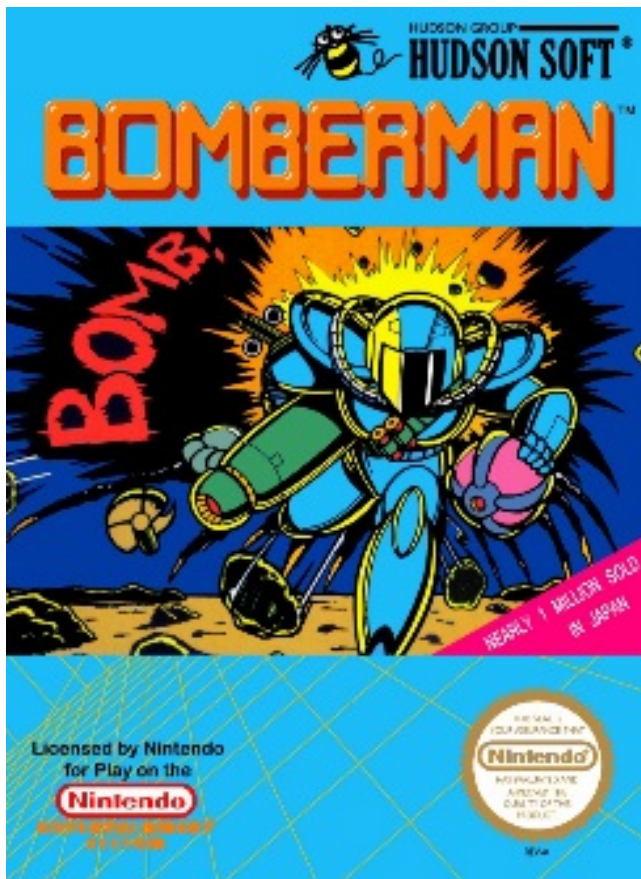


Figure 1: North American NES box art

sicais em valores MIDI e automaticamente retorna a sequência de .word para o uso direto no Assembly.

### 2.2 Personagens

O jogador principal, seguindo a temática proposta, é uma representação da boneca Barbie. O primeiro inimigo introduzido no jogo, que aparece na primeira fase, corresponde ao personagem Bibble, uma referência ao filme "Barbie Fairytopia". O inimigo principal do jogo surge na segunda fase, sendo o Oppenheimer, reforçando o tema de rivalidade entre "Barbenheimer".

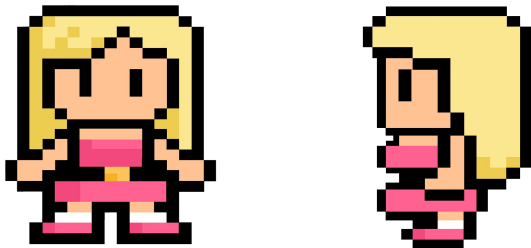


Figure 3: Sprites da personagem principal: Barbie

Para todos os personagens do jogo foram utilizadas sprites de dimensão 16x16. A renderização gráfica dos personagens foi realizada através da determinação de duas coordenadas (x e y), indicando a posição de início da imagem, seguido da renderização de 4 bits por vez até ser obtida a sprite completa. Para a movimentação do jogador, foi utilizada a ferramenta já descrita "Keyboard And Display MMIO Simulator", permitindo a movimentação em quatro direções: esquerda, direita, cima e baixo (teclas A, D, W e S, respectivamente). Para evitar um "rastros" deixado pelos personagens ao serem carregados em locais diferentes do mapa durante a movimentação, foi necessária uma nova renderização da parte do mapa ocupada previamente pelo personagem.



Figure 4: Sprites dos inimigos: Bibble e Oppenheimer

### 2.3 Mapas e colisão

Foram desenvolvidos dois mapas para o jogo, com temáticas diferentes. A primeira fase é uma referência direta à Barbie, com tons predominantes de rosa. A segunda fase, que introduz Oppenheimer, possui uma estética oposta, objetivando um cenário com tons neutros e escuros.

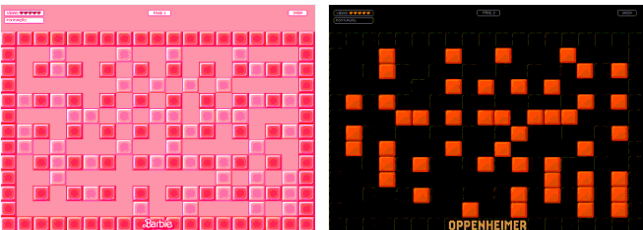


Figure 5: Mapas das fases 1 e 2

Ambos os mapas seguem a proposta do jogo original, com pare-

des fixas intercaladas com blocos destrutíveis por bombas, que permitem uma maior mobilidade do jogador pelo mapa conforme o avanço no jogo. É possível a diferenciação dos blocos destrutíveis e indestrutíveis através da cor dos mesmos.

Para cada um dos mapas, foi criado um mapa correspondente de colisão, como exemplificado na figura 6, onde a cor do byte limitava a movimentação através da sua comparação com a posição do personagem.

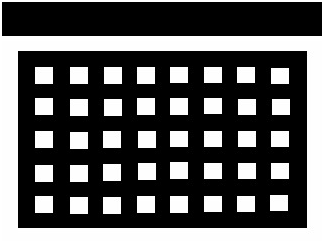


Figure 6: Mapa de colisão

## 3 Resultados Obtidos

O jogo desenvolvido durante esse projeto acompanha a personagem principal "Barbie", buscando derrotar seus inimigos com o uso de bombas, em duas fases distintas. Foram mantidas algumas características fiéis ao jogo original, como a movimentação em quatro direções e o uso de bombas, além da dinâmica de inimigos e condições de vitória das fases (derrotar todos os inimigos).

A primeira fase corresponde à temática do filme "Barbie", com um nível menor de dificuldade. A segunda fase referencia o filme "Oppenheimer", além de contar com uma elevação no nível de dificuldade.

O desenvolvimento do jogo utilizando uma linguagem de baixo nível, provou-se um grande desafio. Foi um projeto trabalhoso e complexo, e foi observada uma dificuldade em encontrar materiais na literatura e na internet que auxiliassem no processo. Nesse sentido, o auxílio dos monitores, e materiais desenvolvidos por outros alunos em períodos anteriores, mostraram-se fundamentais para que fosse possível a implementação desse projeto.

## 4 Conclusão

Durante a realização desse trabalho, foi implementado um novo jogo, inspirado em Bomberman, utilizando-se Assembly RISC-V. O jogo apresenta muitas características objetivadas inicialmente, como duas fases distintas, renderização gráfica dinâmica, movimentação do player e inimigos, entrada a partir do teclado e trilha sonora, porém há a necessidade da implementação futura de novas *features*, como a mecânica de *power-ups*.

## References

DE PAULA, T. T. Livro de oac.

OLDENZIEL, R., 2024. Public History: Introducing Barbenheimer.

SANDERSON, PETE; VOLLMAR, K. Rars – risc-v assembler and runtime simulator.

WIKIPEDIA. Bomberman. Último acesso em: 22 de Julho de 2025.