

Pulsar

(Edição de Halloween)

Caleb Martim
Depto. de Ciências da Computação
Universidade de Brasília
Brasília, Brasil

Gabriel Henrique
Depto. de Ciências da Computação
Universidade de Brasília
Brasília, Brasil

Guilherme da Rocha
Depto. de Ciências da Computação
Universidade de Brasília
Brasília, Brasil

Resumo—Este documento é um relatório para o projeto do grupo 6 da disciplina Introdução aos Sistemas Computacionais do semestre 2022.1 na Universidade de Brasília. O projeto é um jogo de videogame programado em Assembly na arquitetura RISC-V baseado em um outro jogo para fliperamas de 1981, Pulsar.

Pulsar é um jogo de labirinto onde o jogador controla um tanque e seu objetivo é pegar chaves pelo mapa para desbloquear a próxima fase.^[1]

RISC-V, por vez, é um Instruction Set Architecture providenciado sob licenças de código aberto dando a liberdade para qualquer um que desejar projetar um processador baseado nele.^[2]

O jogo do projeto foi desenvolvido com suporte do RARS, um assembler para RISC-V que simula a execução de um programa em um processador RISC-V.

Palavras-chave—RISC-V, Assembly, RARS, Pulsar, Introdução aos Sistemas Computacionais, UnB, jogos, videogame, arcade

I. INTRODUÇÃO

Nossa implementação do jogo tem como objetivo contar uma história fictícia sobre um fantasma que pretende fugir de uma casa assombrada por se sentir fora de lugar nela. O dever do jogador é ajudá-lo a fazer isso, coletando chaves que abrirão portas, enfrentando inimigos, economizando e ganhando combustível, até o fantasma se encontrar finalmente livre. Dessa forma, estamos implementando todas as características do jogo de 1981, Pulsar, apenas com um tema estético diferente.

O projeto tem a intenção de ser um jogo de terror, usando músicas de suspense e sustos, incentivando psicologicamente o jogador a também querer ajudar o fantasma a sair da casa assombrada.

Pela apresentação do projeto acontecer no mês de outubro, nomeamos nosso jogo: Pulsar (Edição de Halloween).



Fig. 1. Banner do jogo Pulsar editado para a nossa versão do jogo

II. METODOLOGIA

A. Música e efeitos sonoros

O jogo possui três músicas, uma de início, quando o jogador inicia o jogo; outra de derrota, quando o jogador morre no jogo; e uma de vitória, quando o jogador vence o jogo.

Cada uma dessas teve que ser manualmente composta usando as ferramentas disponibilizadas no RARS. A partir de syscalls, sons podem ser reproduzidos e dessa forma uma música pode ser simulada, com acordes, ritmo e harmonia.

As músicas em si são todas compostas de sons e músicas culturalmente famosas. A música de derrota é uma derivação do efeito sonoro de suspense famoso "Dun dun duun!" (comumente nomeado assim^[3]) logo a música de vitória é o conjunto dos sete primeiros acordes após a introdução de Marche Funèbre^[4]

```
1 #####
2 # a0: nota #
3 # a1: duração #
4 # a2: instrumento #
5 # a3: volume #
6 #####
7
8 .macro pause
9 li a7, 32
10 li a0, 500 ## Esperar meio segundo antes da próxima nota
11 ecall
12 .end_macro
13
14 .text
15 MAIN:
16     # Primeira metade:
17     li a7, 31 # Tocar som
18     li a1, 1000
19     li a2, 0 # Piano
20     li a0, 40 # E
21     li a3, 127
22     ecall
23     li a0, 59 # B
24     ecall
25     li a0, 64 # E
26     ecall
27
28     pause
29
30     li a7, 31
31     li a0, 59 # B
32     ecall
33     li a0, 64 # E
34     ecall
35
36     pause
37
38     li a7, 31
39     li a0, 39 # D#
40     ecall
41     li a0, 58 # A#
42     ecall
43     li a0, 63 # D#
```

Fig. 2. Exemplo de um dos códigos de música para o jogo

Usamos de auxílio tutoriais de piano disponibilizados no YouTube para que pudéssemos saber as notas e suas durações

de cada música.

Quanto aos efeitos sonoros foi preciso apenas usar o syscall 31 do RARS, onde no registrador a2 é guardado um valor intermediário específico entre 120 à 127. Pois estes são os valores dedicados aos próprios efeitos sonoros que o RARS possui.

B. Arte visual

Seguindo a temática de Halloween, decidimos utilizar uma paleta de cores de tons escuros para denotar o aspecto misterioso e assustador proporcionado pela data comemorativa. Ela foi utilizada na elaboração do mapa da casa assombrada que o jogador deverá escapar. A tela inicial do jogo, por exemplo, é composta por vários tons de marrom para simbolizar o material em que a casa foi construída, no caso madeira, além de trazer a ideia de um local escuro e abandonado.



Fig. 3. Tela inicial que recebe o jogador

Os desenhos dos inimigos foram inspirados por animais comumente encontrados em locais escuros e que as pessoas têm medo. Tendo isso em vista, escolhemos o rato e morcego para implementarmos no jogo. O personagem controlado pelo jogador é baseado na clássica cartoonização de um fantasma, um pano branco com dois olhos pretos.



Fig. 4. Fantasma e seus inimigos

Sabendo que a utilização de cheats na apresentação foi aconselhada, desenhamos um novo design para o jogador que, ao estar sob efeito do cheat, mudaria a cor de seu pano branco para um azul escuro, fazendo assim uma referência ao modo de como os fantasmas do jogo arcade japonês "Pac-Man" (1980) [5] ficavam quando o jogador coletava uma "power pellets" no video-game.

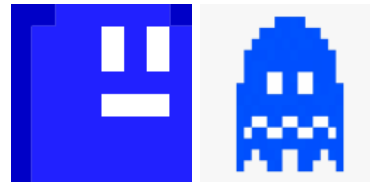


Fig. 5. Fantasma com cheat e fantasma do pacman

Para os tiros, pensamos em ter o fantasma atirar *ectoplasma* como seu ataque contra os inimigos do mapa. O personagem "Geleia" do filme "Os Caça-Fantasmas" de 1984 [6] foi grande parte da inspiração do desing do projétil.



Fig. 6. Sprite do ataque do jogador

Em prol de manter a natureza de um jogo arcade, utilizamos a fonte do jogo "Super Mario World" (1990) (Fig. 7) na cor vermelha como a fonte do nosso projeto, realizando suas devidas adaptações tais como a criação de letras maiúsculas e minúsculas visto que não havia diferença no jogo original.



Fig. 7. Fonte utilizada no jogo Super Mario World

Utilizamos como molde para o mapa do jogo em si o mapa em formato de labirinto presente em "Pulsar". Entretanto, o adaptamos para que a ideia de casa abandonada esteja presente em toda jogabilidade.

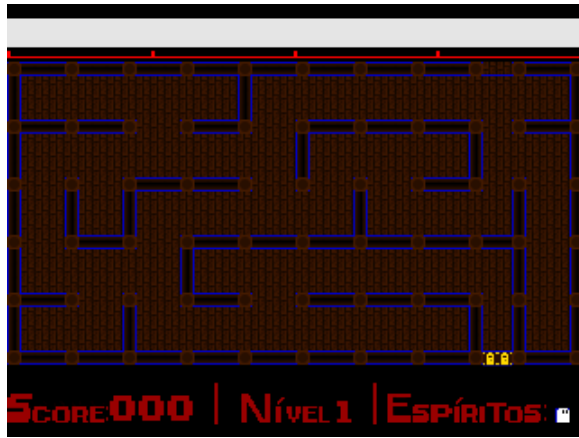


Fig. 8. Modelo do mapa da primeira fase do jogo

C. Lógica do jogo

A base utilizada para o desenvolvimento das mecânica que foram utilizadas no jogo foi a estrutura do diagrama (Fig. 9) que criamos para facilitar a direção do desenvolvimento.

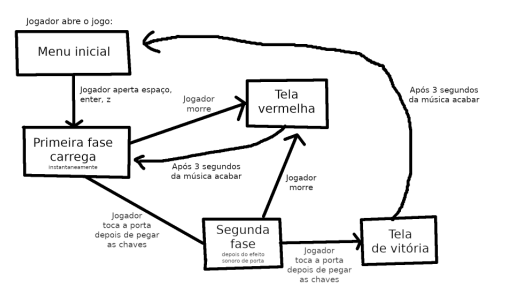


Fig. 9. Diagrama da estrutura do nosso jogo

Por meio do diagrama, foi possível desenvolver a estrutura básica para a troca entre as diferentes etapas do jogo, que são: Tela de Início, Fase 1, Fase 2, Tela de Morte e Tela de Vitória.

Outra mecânica importante definida a partir do diagrama foram as condições que levariam para as mudanças entre as diferentes etapas, o que guiou o desenvolvimento do trabalho.

Para o desenvolvimento das mecânicas internas do jogo, as quais não foram claramente definidas no diagrama base, foram desenvolvidos diferentes métodos para atingir os objetivos desejados. As mecânicas, com seus respectivos métodos, foram exemplificadas nos próximos tópicos.

1) *Sistema de colisão*: O sistema de colisões e coleta de chaves foi baseado na análise das cores dos pixels nos arredores dos personagens, tiros e inimigos. Comparando as diferentes cores possíveis tornou-se possível determinar as reações esperadas para cada um dos casos e tornou um único código capaz de atender todos os objetivos citados.

Tendo a figura 10 como exemplo do funcionamento desse código, podemos observar que a parede possui uma cor específica em sua volta que foi definida como o código hexadecimal 0xC7, a qual é interpretada pelo RARS como um pixel invisível. Fazendo uso dessa cor apenas para as paredes

e o personagem tornou-se possível detectar quando teria uma colisão com objetos que não possuíam reação.

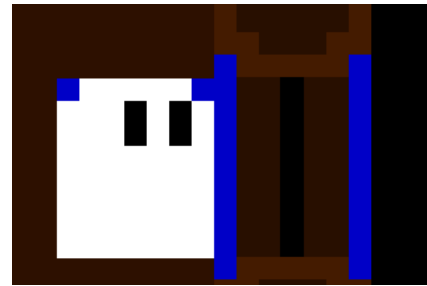


Fig. 10. Representação do sistema de colisão entre o personagem e as paredes do jogo

Além disso, o mesmo método foi usado para determinar a colisão de tiros com inimigos, coleta de chaves, passagem de fase e danos causados por inimigos, cada um fazendo uso de uma cor distinta para facilitar a interpretação e evitar possíveis ambiguidades no código.

2) *Movimentação do personagem*: A movimentação do personagem, dos inimigos e dos tiros e a limpeza de rastros foram baseados em labels que continham os valores da posição anterior e atual dos alvos da análise. As labels contêm os valores X e Y de cada objeto que se move no jogo, os quais podem ser acessados por outras partes do código, possibilitando o uso dessas informações não só para a movimentação do personagem como também para a limpeza de rastros e o funcionamento do sistema de colisões.

3) *Inteligência dos inimigos*: A inteligência dos inimigos criados para o jogo foi baseada no sistema de colisão, no sistema de movimentação do personagem e no sistema de tiros. Por meio da junção desses sistemas foi possível criar uma imagem que se move em uma direção determinada e troca seu curso baseado nas colisões que sofre.

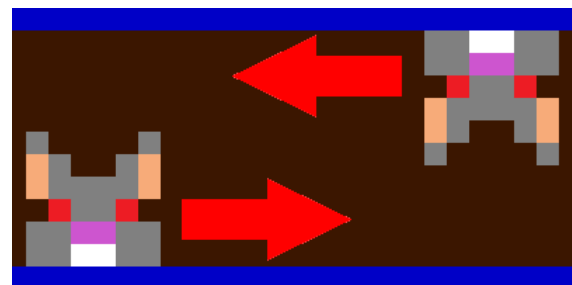


Fig. 11. Ilustração de movimento do inimigo rato

o movimento dos ratos é em um único eixo, podendo ser para direita ou para esquerda. Esse mesmo conceito foi utilizado para os demais inimigos empregados no jogo.

4) *Pontuação do jogador*: No sistema de score foi utilizado uma label para acumular toda a pontuação acumulada ao longo do jogo. Para a representação gráfica, foi utilizado o resto da divisão do valor da label para determinar as centenas, dezenas e unidades que deveriam ser exibidas na tela, e com o uso

de branches foi possível determinar quais dígitos deveriam ser exibidos.



Fig. 12. Exemplo da pontuação do jogador

III. RESULTADOS OBTIDOS

No geral, nossas dificuldades foram superadas e conseguimos criar um jogo completo, com início, meio e fim, apesar de ainda conter alguns erros de jogabilidade.

O tema que tínhamos em mente foi implementado com sucesso assim como a maioria das técnicas que elaboramos para a execução do projeto.

Como resultado, a equipe adquiriu um melhor entendimento das mecânicas para o desenvolvimento em Assembly e a lógica de programação em Assembly na arquitetura RISC-V, assim como noções de desenvolvimento em baixo nível, como gestão manual de memória e planejamento de um projeto com recursos limitados.

Apesar de não termos conseguido implementar uma música de fundo para tocar durante o jogo, nós conseguimos implementar várias músicas, cada uma para situações diferentes no jogo, melhorando a experiência de jogador.

Em relação a arte visual do jogo, esta foi bem executada, conseguimos criar um sprite^[7] para todos os objetos e personagens do jogo.

Alguns dos detalhes técnicos que tínhamos como planejamento não foram implementados perfeitamente. Por exemplo, relatamos que o sistema de pontuação é algumas vezes instável.

Outra coisa que não funciona idealmente é o sistema de eliminação de inimigos. Ao jogador atirar nos inimigos, o jogador ganha combustível, assim como deveria ser feito, porém eles não morrem. A morte dos inimigos só ocorre quando o personagem principal passa na frente deles. Dessa forma, a morte dos inimigos ocorre, porém não exatamente como planejamos.

CONCLUSÃO

Tendo em vista os pontos observados, Pulsar (Edição de Halloween) é um projeto criado por nosso grupo contendo os elementos do jogo de arcade de 1981, Pulsar, com temas de Halloween no intuito de ser um jogo de terror.

Mesmo com recursos bastante limitados, conseguimos criar algo que pode ser considerado um jogo finalizado. O jogo tem um objetivo, possui arte gráfica, sonora e diversas técnicas para a jogabilidade ser possível.

Esperamos que quem jogar tenha uma boa experiência e possa se divertir. Feliz Halloween!



Fig. 13. Tela final do jogo, quando o fantasma consegue fugir da casa assombrada

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] https://www.arcade-museum.com/game_detail.php?game_id=9150
- [2] <https://riscv.org/about/>
- [3] https://en.wikipedia.org/wiki/Dun_dun_duuun!#CITEREFSuspense1942
- [4] https://youtu.be/hZY5DBmgC_A
- [5] <https://pt.wikipedia.org/wiki/Pac-Man>
- [6] <https://youtu.be/mDYdtTuFUKw>
- [7] [https://pt.wikipedia.org/wiki/Sprite_\(computa%C3%A7%C3%A3o_gr%C3%A1fica\)](https://pt.wikipedia.org/wiki/Sprite_(computa%C3%A7%C3%A3o_gr%C3%A1fica))