

Análise do Jogo Centipede (em Bash)

INTRODUÇÃO

- Durante a aula do dia 15/08, foi disponibilizado o código do jogo Centipede programado em Bash.
- O jogo abriu mas não rodou corretamente em minha máquina: (Windows 8.1 no terminal Git Bash):
- A Centipede não aparecia e logo após dava a mensagem de fim de jogo.

DESENVOLVIMENTO

- **Com ajuda da IA ChatGpt:**
 - ✓ Analise do código original,
 - ✓ Identificação dos pontos de incompatibilidade / inconsistência,
 - ✓ Realização dos ajustes e testes consecutivos até conseguir uma versão funcional no meu ambiente.
 - ✓ Implementação de melhorias e novas funcionalidades que tornaram o jogo mais completo e agradável:

- **Linha de Status do Jogo (Acima da borda superior)**

Além do campo SCORE (implementado no código original do jogo), que marca a pontuação, foram acrescentados:

- ✓ Campo (FPS): Faz a contagem dos Quadros por Segundo,
- ✓ Campo (Timer): Faz a contagem regressiva de 5 minutos, sendo resetado sempre que o jogo reinicia.
- ✓ Mensagens: "Game-Over" ao perder a partida.
 - Aparecia apenas "Game Over" em uma linha perto do rodapé da tela.
 - Em geral, sem destaque, apenas texto simples.
 - jogo terminava logo em seguida, sem opção de reinício (só fechava).

Game Over → fim direto.

Novas Implementações e Mensagens mais completas (centralizadas na Tela do Jogo):

Quando o jogador perde:

- A mensagem aparece no centro da tela:
- GAME OVER
- Aperte R para reiniciar
- Aperte S para sair
- Texto normal (sem símbolos estranhos).
- O jogador pode escolher reiniciar ou sair, sem precisar fechar o terminal ou executar a linha de comando para uma nova partida.

Quando o jogador ganha (Timer chega a 0)

- Aparece mensagem de vitória também centralizada:
- VOCÊ VENCEU!
- Aperte R para reiniciar
- Aperte S para sair
- Timer (que começa em 5 minutos, por exemplo) é o critério para vitória.

Existem dois finais possíveis:

Derrota: colisão → “Game Over” .
Vitória: tempo zerado → “Você Venceu!” .

Observação: Com relação a vitória pode ser incluído uma nova condicional, ou seja, além de ter que permanecer vivo nos 5 minutos, garantir um SCORE mínimo de 80 pontos.

- **Resumo das Correções e Melhorias do Jogo Centípede:**

Aspecto	Código Original	Código Final (Corrigido)	Resumo da Correção
Cores	tput setf / setb (incompatíveis)	tput setaf / sgr0 (funciona no Git Bash)	Adotou-se tput setaf/sgr0, com índices válidos e reset claro → cores estáveis e portáteis, resolvendo o problema visual.
Caracteres	Especiais (\0256, \0244, \0362)	ASCII (@, X, o)	Caracteres especiais (\0256, \0244, \0362) não apareciam no terminal do Windows. Os caracteres especiais foram substituídos por caracteres ASCII simples e garantidos, compatíveis com praticamente qualquer terminal.
Loop do jogo	trap move ALRM + multiplos processos paralelos	Loop único em while	Processos paralelos → loop único. O loop foi simplificado para rodar em um único processo, garantindo que o jogo seja estável, fluido e compatível com o Git Bash do Windows.
Direção inicial	Aleatória (podia gerar Game Over)	Fixa: direita	Aleatório → Direção fixa inicial. A direção inicial foi fixada para a direita (DIRECTION="r"). A centípede começa sempre indo para a direita, evitando colisão logo no início. Garante tempo de reação para o jogador antes de perder a partida.
Entrada de teclas	Apenas h j k l	h j k l + setas do teclado	Somente hjkl → Suporte também às setas
Game Over	Mensagem no rodapé e saída	Tela centralizada + opções R (reiniciar) / S (sair)	Rodapé simples → Centralizado com opções

Manipulação de arrays	echo cut (instável no Windows)	Bash arrays nativos	cut/echo → Arrays nativos do Bash. Manipulação da centipede passou de echo cut (instável) para arrays Bash nativos, trazendo estabilidade no Windows. Garantia de movimentação fluida e confiável.
Status do jogo	Apenas SCORE simples	SCORE + FPS + TIMER, centralizados no topo	Só SCORE → SCORE + FPS + TIMER
Timer	Não existia	Regressivo (5 min), vitória ao terminar	Sem timer → Timer regressivo
FPS	Não existia	Cálculo por segundo	Sem FPS → FPS calculado. Adição do cálculo de FPS por segundo, inexistente no original, permitindo acompanhar fluidez e desempenho.
Compatibilidade	Linux apenas	Funciona no Git Bash (Windows 8.1)	Linux restrito → Compatível com Windows 8.1 e Terminal Git Bash

CONCLUSÃO:

Embora o objetivo da prática “consistente” deste jogo visa o controle da Centipede por meio do teclado, a intensão por trás é proporcionar, aos alunos, maior afinidade com os comandos do Vim ou NeoVim. Apesar disso, a incompatibilidade do Código original com meu notebook, me trouxe a oportunidade de praticar a utilização da Inteligência Artificial no apoio a identificação de inconsistências no Código, assim como a elaboração dos novos recursos aplicados, tudo isto, através da interação com o **prompt de commando**.