

Universidade Federal da Fronteira Sul

GEX1213 – Organização de Computadores

## **Relatório Técnico**

### **Jogo Mancala em Assembly RISC-V**

**Integrante:**

João Luís Almeida Santos – 20240002408

Chapecó – SC

5 de novembro de 2025

## Conteúdo

<b>I. Introdução</b>	<b>2</b>
Demanda do Trabalho . . . . .	2
<b>II. Explicação do Código</b>	<b>2</b>
Inicialização . . . . .	2
<b>III. Conclusão</b>	<b>3</b>

## I. Introdução

Este relatório descreve o desenvolvimento de uma implementação do jogo Mancala em Assembly RISC-V, executada no simulador RARS, como parte das atividades da disciplina de Organização de Computadores. A proposta do trabalho foi de simular o jogo de tabuleiro Mancala em formato de terminal.

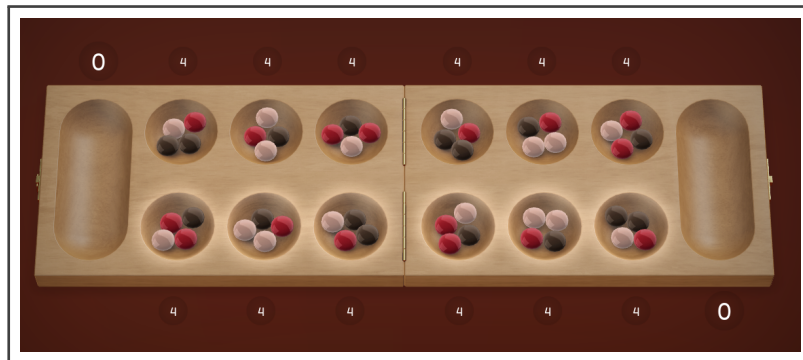


Figura 1: Representação do tabuleiro do Mancala (CrazyGames)

### Demanda do Trabalho

O enunciado disponibilizado exige uma versão do Mancala com doze cavidades e dois poços, quatro sementes por casa no estado inicial, suporte a turnos extras, captura de sementes e detecção do fim de jogo. Dentro do código, procurei modularizar e abstrair o máximo da lógica, dado o uso de funções de Macro e a criação de funções que vaziam processos simples, como printar, printar em loop, ler inteiro, etc.

## II. Explicação do Código

### Inicialização

Os primeiros passos do programa são dados na seção .data. Lá, são declaradas as variáveis, textos necessários para as funções de print, a vitória do jogador, o turno atual, etc. Além disso, todas as cavidades são inicia-

lizadas com o valor de 0, e a variável **SEED\_INIT** é criada com o valor 4. Esta variável pode ser alterada para mudar a funcionalidade do jogo.

```
1  SEED_INIT:
2  .word      4
3
4  vitorias_j1:
5  .word 0
6  vitorias_j2:
7  .word 0
8  turno_atual:
9  .word 0
10 cavidades:
11 .word      0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
12 SEED_INIT = 4;
```

### III. Conclusão