

Universidade Federal de Santa Catarina - UFSC
Departamento de Informática e Estatística - INE
Curso de Ciências da Computação
INE5417 - Engenharia de Software I

Especificação de Requisitos: Jogo Chaturanga

Alunos:

André Filipe da Silva Fernandes
Eduardo Vicente Petry
Gabriel Holstein Meireles

Professor:

Ricardo Pereira e Silva

Florianópolis

2022

1 Introdução

Versão 0.1 16/05/2022

Versão	Autores	Data	Ação
0.0	André Filipe da Silva Fernandes, Eduardo Vicente Petry, Gabriel Holstein Meireles	09/05/2022	Estabelece os requisitos iniciais
0.1	André Filipe da Silva Fernandes, Eduardo Vicente Petry, Gabriel Holstein Meireles	16/05/2022	Corrige imagens e realoca seções

1.1 Objetivo do Desenvolvimento

Desenvolvimento de um software para computador que permita a disputa local entre dois jogadores em uma partida de Chaturanga, um antigo jogo indiano.

1.2 Descrição do Jogo

Chaturanga é um jogo composto por dois jogadores, um tabuleiro 8x8 e 16 peças para cada jogador. As 16 peças compõem um exército e se dividem em oito Padati, dois Ratha, dois Ashwa, dois Gaja, um Mitri e um Raja.

A distribuição do exército no tabuleiro para o início da partida se dá como na imagem 1.

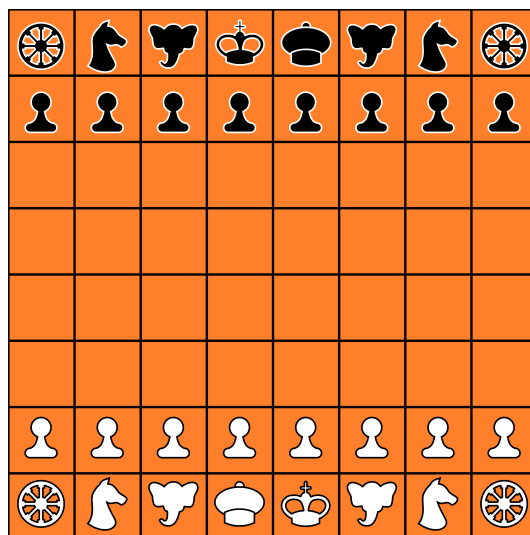


Figura 1: Configuração Inicial do Tabuleiro

O jogador com as peças brancas deve iniciar a partida e, então, os turnos alternam entre os dois jogadores. O objetivo do jogo é capturar o Raja inimigo ou aniquilar o exército adversário.

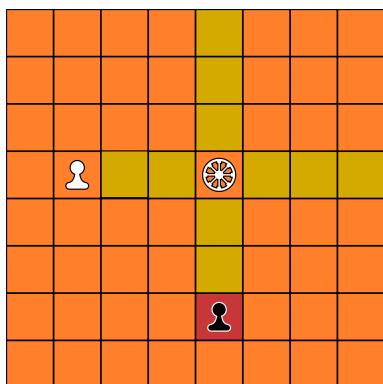
A cada turno, o jogador escolhe uma peça de sua cor que está no tabuleiro e a move para alguma casa vazia ou captura uma peça adversária.

As regras de movimento para cada peça são as seguintes:

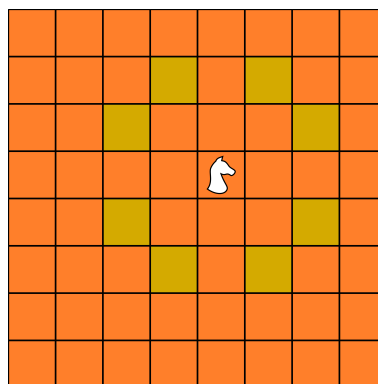
- Padati (Soldado): anda uma casa diretamente para frente ou, no caso de existir uma peça inimiga na sua diagonal direta, captura uma peça inimiga em sua diagonal direta, como na figura 2.
- Ratha (Carruagem): anda em movimento ortogonal ao tabuleiro, ou seja, tanto para frente/trás quanto para esquerda/direita e se move quantas casas forem desejadas, como na figura 2.
- Ashwa (Cavalo): anda imediatamente na diagonal oposta de um retângulo 3x2 e ignora qualquer peça que exista no caminho, como na figura 2.
- Gaja (Elefante): anda duas casas em uma diagonal adjacente e ignora qualquer peça que exista no caminho, ou uma casa diretamente para frente, como na figura 2.
- Mitri (Ministro): anda uma casa em uma diagonal adjacente, como na figura 2.
- Raja (Rei): anda uma casa em qualquer direção adjacente, como na figura 2.

Nota-se que no caso das peças que tem movimento de várias casas, elas param ao colidir com uma peça aliada ou ao capturar uma inimiga a não ser que seja especificado que ignoram outras peças. Em caso de movimento que resulta em captura, a peça que fez o movimento passa a ocupar a casa previamente pertencente a capturada.

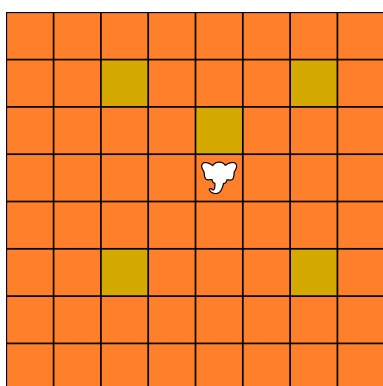
No caso especificamente do Padati, há a possibilidade de promoção para virar um Mitri. Para isso, é necessário que a peça encoste na borda do tabuleiro pertencente ao exército do adversário.



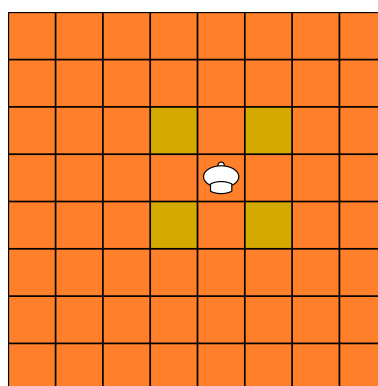
(a) Movimentos do Ratha



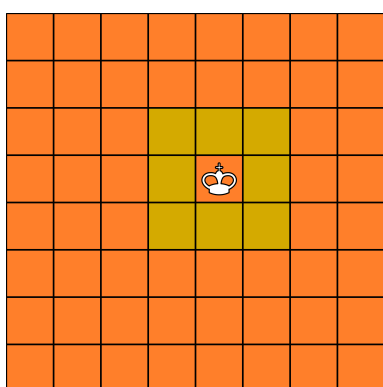
(b) Movimentos do Ashwa



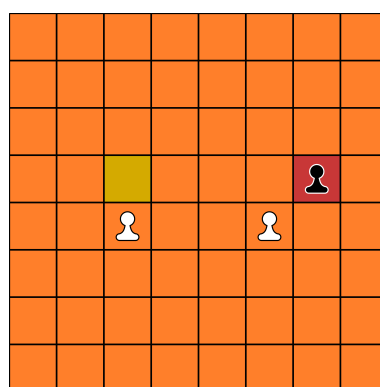
(c) Movimentos do Gaja



(d) Movimentos do Mitri



(e) Movimentos do Raja



(f) Movimentos do Padati

Figura 2: Movimentação das peças

1.3 Referências

<https://www.chessvariants.com/historic.dir/chaturanga.html>
<https://en.wikipedia.org/wiki/Chaturanga>

2 Visão Geral do Sistema

2.1 Arquitetura do Programa

Programa orientado a objetos escrito na linguagem Python.

2.2 Premissas de Desenvolvimento

- O programa será desenvolvido com a linguagem de programação Python 3 sob o paradigma de Orientação a Objetos.
- O programa deverá apresentar uma interface bidimensional, onde cada jogador irá interagir com o uso do mouse.
- O jogo será implementado em português.
- Uma partida será iniciada junto com a execução do programa.

3 Requisitos de Software

3.1 Requisitos Funcionais

- Requisito Funcional 1 - Reiniciar partida: no turno do jogador, ele poderá clicar em um botão escrito "Reiniciar partida" para reiniciar o jogo. Caso o botão seja clicado, será considerado vitória do jogador adversário e o tabuleiro retornará para a configuração inicial.
- Requisito Funcional 2 - Escolher peça: no turno do jogador, ele poderá clicar em uma peça qualquer que está no tabuleiro. Ao clicar na peça, as possíveis casas para qual tal peça pode se mover devem tornar-se amarelas. Após uma peça ser escolhida, só poderá ser possível escolher outra peça no mesmo turno após a execução do Requisito Funcional "Mover peça". Nota-se que cada peça possui movimentos distintos dependendo do seu tipo e isso deve ser obedecido na hora de determinar a posição das casas amarelas.
- Requisito Funcional 3 - Mover peça: no turno do jogador, após escolher uma peça, o jogador poderá movê-la para uma das casas amarelas ao clicar nela e capturar uma peça adversária se a casa escolhida estiver ocupada por ela. Caso seja clicado em qualquer casa após uma peça ser escolhida, ela deixará de ser a peça escolhida. Assim, se foi clicado em uma casa

não amarela e não houve movimento, será necessário escolher uma peça novamente para realizar o movimento. Sempre que uma peça é movida, deve-se verificar se a condição de vitória foi atingida conforme a Descrição do jogo e, se sim, a partida é finalizada e o jogador da vez é o vencedor. Se não, o turno é finalizado e torna-se a vez do jogador adversário.

3.2 Requisitos Não Funcionais

- Requisito Não Funcional 1 - Especificação de Projeto: Além do código em Python deve ser produzida especificação de projetos baseada em UML2, através da ferramenta Visual Paradigm.
- Requisito Não Funcional 2 - Interface Gráfica: O programa deve implementar a interface gráfica utilizando a biblioteca Pyqt5. Como na imagem [3](#)



Figura 3: Esboço da Interface