

Jogo de Ludo

Cada jogada ocorre na seguinte forma:

1. O jogador joga um dado e, em seguida, escolhe uma das peças para avançar no sentido horário ou para frente caso ela esteja na reta final

Abrigo (O local em que todas as peças do jogador ficam no início da partida)

2. Para sair do abrigo, é necessário tirar 1 ou 6 no dado.
3. Ao sair do abrigo, se o jogador tirou 6, ele pode jogar o dado novamente e andar com qualquer peça.
4. Se ao sair do abrigo houverem peças de jogadores adversários na casa de origem, uma captura ocorre e elas retornam para o abrigo do adversário.
5. Caso não queira ou não possa retirar uma peça do abrigo, o jogador deve escolher uma peça que não está nele para andar o número de casas que saiu no dado.
6. A peça só pode andar se não houver nenhuma obstrução no caminho (inclusive as feitas pelo próprio jogador) dela considerando todos os passos que ela deve fazer.

Uma obstrução é quando duas peças estão na mesma casa.

Por exemplo, se eu quero andar 4 casas com uma peça, mas tem uma obstrução na 3ª casa, somente andarei 2 casas.

Se a obstrução está na última casa da jogada, ambas as peças que estavam na obstrução são comidas e voltam para a origem.

7. Se o jogador terminar o turno em cima de uma casa que já abriga uma peça inimiga, a peça inimiga retorna para o abrigo do adversário, é "comida".

Final

8. Ao andar 51 casas o jogador pode entrar na reta final correspondente a sua cor.
9. Na reta final o jogador precisa tirar no dado o número exato de casas que faltam para chegar ao fim para ganhar. Se o jogador tirou um número menor que esse, ele pode andar normalmente, se for maior ele anda até o fim e volta a quantidade restante.
10. Se todas as peças de um jogador completarem a reta final, ele é o vencedor.

Requisitos Funcionais

Sobre o jogo

Podem jogar de 2 a 4 jogadores em um tabuleiro cada um com uma cor dentre: Vermelho, Amarelo, Verde ou Azul.

Tabuleiro composto por casas que podem conter até duas peças.

O tabuleiro tem 52 casas além de uma Reta Final por jogador, que possui 5 casas cada.

O circuito normal, composto pelas 52 casas é sempre percorrido no sentido horário.

A reta final é alcançada após o jogador completar o circuito normal.

Cada jogador tem uma cor, 4 peças, um abrigo, uma casa de origem e uma de saída. Um jogador joga por turno. Em um turno o jogador joga um dado, anda com a peça que ele escolher levando em conta obstruções e se ele está fora do abrigo ou não, entre outras restrições. Se um jogador colocar sua peça na mesma casa que tem uma ou duas peças do outro jogador, a(s) peça(s) do outro jogador volta(m) para o abrigo dele.

Ganha o jogador que colocar as 4 peças no final.

Módulo Lista Duplamente Encadeada

- Implementar uma lista duplamente encadeada.
- Todas as funções de manipulação incluindo inserção, busca e outras.
- Deve conseguir operar mais de uma lista.

Módulo Lista Circular

- Deve implementar uma lista encadeada onde o último elemento sempre aponta para o primeiro.
- Todas as funções de manipulação da lista como avançar, inserir, procurar, etc.
- Deve conseguir operar mais de uma lista.

Módulo Desenho

- Representar o tabuleiro com suas casas e peças de uma forma visual e intuitiva.
- Permitir identificação fácil da rodada atual e de todas as opções de peças que o jogador pode mexer.

Módulo Partida

- De 2 a 4 jogadores
- As jogadas são feitas por um jogador depois do outro, seguindo sempre a mesma ordem
- Capaz de controlar todos os outros módulos a fim de executar uma partida do jogo Ludo do início ao fim.
- ▶ Responsável por:
 - avisar ao tabuleiro que uma nova jogada de determinado jogador deve se iniciar
 - Iniciar e controlar a partida.
 - Avisar quando tiver um vencedor

Módulo Tabuleiro

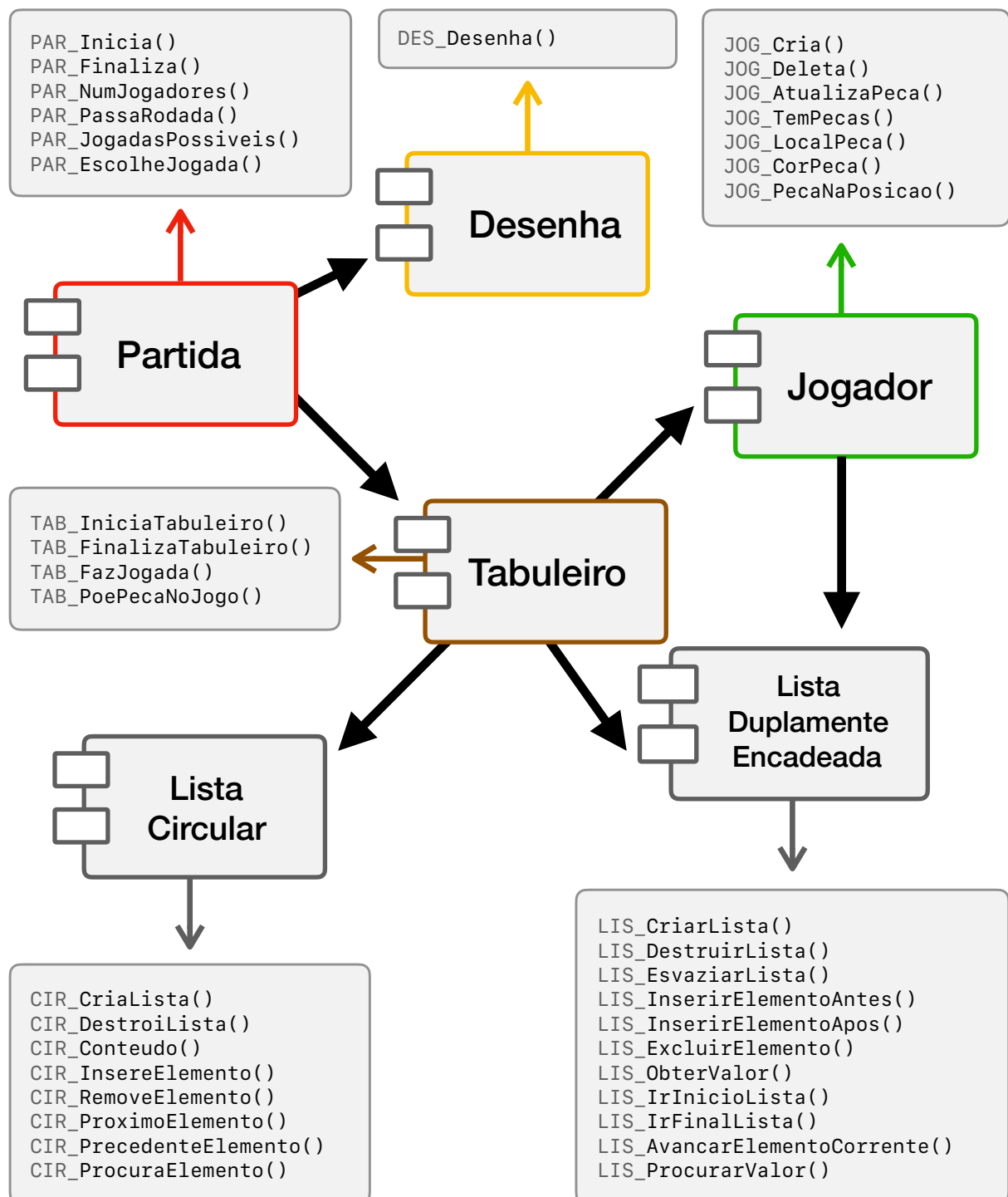
- Implementar a estrutura casa, que pode conter 2 peças além de poder ter atributos como cor ou uma bifurcação para uma reta final.
- 52 casas
- 4 retas finais com 5 casas cada
- ▶ Responsável por:
 - Fazer o procedimento das jogadas.
 - Analisar e avisar se a jogada é válida.
 - Avisar se uma peça chegou ao final.

Módulo Jogador

- Jogador que contém 4 peças e uma cor.

- Peças que guardam sua posição e sua cor
- Posição de origem das suas peças
- Cor

Visualização das dependências



Requisitos não-funcionais

Manutenibilidade

Os módulos serão feitos com responsabilidades bem-definidas em mente. Todos eles terão suas funcionalidades checadas por meio de testes individuais, utilizando as ferramentas fornecidas pelo arcabouço de testes. Eles só serão adicionados ao resto do programa após um teste de integridade usando a ferramenta mencionada.

Consistência

Todos os módulos serão compostos utilizando os mesmos padrões de programação focando em documentar explicitamente coisas como parâmetros recebidos pelas funções, retornos esperados e explicitação da abstração que cada módulo representa. Essas regras simplificam a leitura e o entendimento dos módulos facilitando sua integração e auxiliando na sua manutenção.

Robustez

O módulo tabuleiro deve validar cada jogada e avisar para o jogador caso seja ilegal. Desse modo qualquer dado de entrada inválido será parado por esse módulo antes de poder causar outros erros.

Corretude

O jogo será desenvolvido com ferramentas que facilitem o teste utilizando injeção de dependência entre outras técnicas a fim de possibilitar a simulação controlada de situações de jogo reais. Incluiria utilidades como lançamento de dados não aleatório, inserção de peças arbitrária, controle manual de rodadas, etc.

Eficiência

O jogo será montado de forma eficiente, sem códigos excessivamente redundantes como verificações desnecessárias ou loops maiores que o suficiente.

Usabilidade

O programa final será intuitivo de usar e de fácil entendimento. Sem assumir que o usuário estará ciente de como o código funciona internamente, apenas requerindo que o mesmo saiba as regras do Ludo.