

**Disciplina: Programação Orientada a Objetos 2024.1**

**Professora: Beatriz Trinchão Andrade**  
**Colaboração: prof. André Britto e Mislene Nunes**

**Jogo Cata-Frutas**

**1. Introdução**

Neste trabalho vocês farão um jogo voltado ao acúmulo de frutas. A interface será uma área de duas dimensões, vista de cima, que representa uma floresta. A floresta será composta por elementos, que podem ser pedras, árvores, e grama. Dentro da floresta, teremos competidores e frutas, que fazem parte da dinâmica do jogo.

**2. Funcionamento básico do jogo**

O jogo funciona da seguinte forma: dois competidores entram na floresta, e vence aquele que conseguir primeiro mais da metade das frutas ouro, onde  $x$  representa a quantidade de frutas ouro fornecida na inicialização do jogo.

O jogo é composto por:

**Uma floresta:** nossa floresta é bidimensional, e é representada como uma matriz quadrada de tamanho  $m \times m$ . Cada posição da floresta é ocupada por um elemento estático.

**Elementos:** os elementos podem ser estáticos ou dinâmicos.

- **Elementos estáticos:** são aqueles que não mudam de lugar ao longo do jogo. São eles: pedras, árvores e grama;
- **Elementos dinâmicos:** são aqueles que mudam de posição ao longo do jogo: competidores e frutas. O competidor percorre a floresta à procura de frutas. A fruta sofre mudança de posição ao decorrer das ações do competidor, ou seja, quando um competidor cata a fruta da grama e a armazena em sua mochila, ao mudar a sua posição também estará mudando a posição da fruta. Além disso, a fruta pode cair da mochila e ocupar outra posição na floresta, assim como também pode ser consumida pelo competidor e, portanto, sumir da floresta.

O jogo começa com 2 competidores entrando na floresta e termina quando o primeiro deles conseguir acumular  $x/2 + 1$  frutas ouro, onde  $x$  representa a quantidade de frutas ouro fornecida na inicialização do jogo. Os competidores devem se movimentar pela floresta. O movimento é definido pela mudança da localização do competidor de uma posição da floresta para uma posição adjacente (ver Seção **Movimento**).

Cada competidor se movimenta por vez. A ação é sempre resolvida após o movimento do competidor da vez, ou seja, antes da jogada de seu adversário. O lançamento de dois dados irá fornecer o número de **pontos** de movimento que podem ser utilizados no percurso. Não é preciso que o jogador gaste todos os pontos obtidos em uma rodada. No entanto, os pontos não são cumulativos, ou seja, são zerados ao final de cada rodada.

Toda vez que um competidor se movimenta, podem ocorrer as seguintes ações:

- **competidor entra em uma célula ocupada por grama:** o competidor passa a ocupar a célula escolhida pelo jogador ao custo de **um ponto**.
- **competidor entra em uma célula ocupada por uma árvore:** o competidor ocupa a célula correspondente ao custo de **um ponto**. Ao decidir não se mover na próxima rodada, uma fruta da árvore é garantida. Esse benefício só é possível uma vez a cada 5 rodadas - não adianta ficar para sempre na sombra da árvore esperando as frutas caírem. As árvores de diferentes tipos são alocadas automaticamente no início do jogo.
- **competidor se move para uma célula ocupada por uma pedra:** o competidor não pode permanecer nesta célula, podendo apenas pular a pedra ao custo de **três pontos** (um ponto é necessário para chegar na célula, outro ponto para poder pular a pedra, e por fim, um ponto para avançar para a próxima célula). A pedra deve continuar na mesma célula. Caso não consiga pontuação suficiente para este movimento, o competidor deve permanecer na célula que estava anteriormente. As pedras são alocadas aleatoriamente no início do jogo.
- **competidor se move para uma célula grama ocupada por uma fruta:** o competidor pode carregar a fruta e, portanto, ficar mais forte. O custo deste movimento é igual ao da grama, ou seja, **um ponto**. A fruta deve ser movida da floresta para a mochila do competidor. As frutas de diferentes tipos são alocadas aleatoriamente no início do jogo (mais sobre isso à frente).
- **competidor tenta entrar em uma célula ocupada por outro competidor:** agora vem a confusão! Toda vez que dois competidores se encontram, eles têm algum desentendimento. A confusão é sempre unilateral, ou seja, o competidor da vez enfrenta o outro no empurrão. Quando um competidor empurra o outro, os dois competidores permanecem na mesma posição do momento em que a ação foi feita (em células adjacentes). As frutas do competidor empurrado são derrubadas de sua mochila, e distribuídas em volta do jogador empurrado. Caso ainda tenha pontos de movimento, o competidor que empurrou pode catar frutas que são derrubadas da mochila do seu adversário. Quanto mais frutas um competidor tiver na mochila, mais forte ele será.  
Obs. Um jogador não pode ocupar a mesma posição do outro, ou passar por uma célula ocupada pelo adversário.

O competidor também pode consumir uma fruta, ao custo de um ponto de movimento.

A quantidade de frutas que cai no chão depende da força de ataque daquele que está empurrando ( $f_a$ ) e da força de defesa do adversário ( $f_d$ ). A força de cada competidor depende da quantidade  $n$  de frutas que sua mochila contém. Ou seja:

$$f_a = n_a$$

$$f_d = n_d$$

Caso o competidor atacante resolva usar uma fruta de força no momento do empurrão, a sua força dobrará sobre as frutas resultantes na mochila, ou seja:

$$f_a = 2*(n-1)$$

Apenas uma fruta de força pode ser usada por vez. O competidor atacado não pode usar fruta de força para defesa. Além disso, só pode haver um ataque por jogada.

A quantidade de frutas que cairá da mochila do competidor empurrado é calculada através da equação a seguir:

$$\text{quantidade de frutas derrubadas} = \max(0, \text{empurrao})$$

onde:

$$\text{empurrão} = \text{round}(\log_2(f_a+1)) - \text{round}(\log_2(f_d+1))$$

Caso o competidor empurrado tenha menos frutas que a quantidade de frutas descrita na equação acima, ele deve derrubar todas as frutas que tem em sua mochila.

### 3. Movimento

Na sua vez no turno, o competidor deve definir seu caminho pela floresta. A perspectiva do jogo é a floresta vista de cima, assim, o competidor sabe a localização de todos os elementos do jogo.

O caminho é definido pelo jogador, através do controle via teclado ou mouse. O número de movimentos que ele poderá fazer em uma rodada é limitado pela pontuação obtida com os dados e pelos custos das casas por onde o competidor passará.

Na sua vez de jogar, o competidor pode se mover para células da vizinhança-4. Ou seja, se ele está na posição X na figura abaixo, suas opções são as células 1, 2, 3 ou 4.

	1	
2	X	3
	4	

Cada movimento consumirá um ou mais pontos, a depender do que esteja na posição escolhida. Ao fim destes pontos, o jogador não se move mais naquela rodada. O jogador também pode optar por parar antes de gastar os pontos. Neste caso, deve haver um botão na interface que indique que o jogador quer concluir sua jogada. Ao escolher esta opção, seus pontos de movimento são zerados e a vez é passada para o outro jogador.

Comandos: via teclado, o jogador deve pressionar as setas direcionais para indicar a próxima posição do competidor na vizinhança-4. Por mouse, ele deve clicar em uma posição na vizinhança-4. Caso ele selecione uma posição inválida ou seus pontos não permitam o movimento desejado, o jogo deve emitir um aviso.

### 4. Inicialização

O terreno é gerado com base nas configurações do usuário. Essas configurações devem ser definidas e salvas em arquivo, para que o usuário possa carregar uma configuração específica quando quiser.

Os valores configuráveis são:

- A dimensão  $m$  da floresta ( $m \geq 3$ ).
- O número de pedras (o restante da floresta é ocupada por grama).
- O número total de frutas ouro (deve ser no mínimo 1).
- O número de cada tipo de fruta não-ouro que estará no chão no início do jogo (agilidade, força, antídoto, ou sem propriedade extra) e os tipos de árvores desejados para aquela partida e a quantidade de árvores de cada tipo. A quantidade não pode ultrapassar o limite de células com grama. Não existem árvores de maracujá - a planta cresce se apoiando em outras (mais sobre isso na seção “Vitória”).
- A chance de uma fruta estar bichada (até as frutas ouro podem estar bichadas).

- Capacidade da mochila do competidor - em número de frutas.

Arredondar o número de frutas obtido, em caso de valores não inteiros.

No início do jogo, uma quantidade definida de frutas ouro é colocada em posições aleatórias, ocupadas por grama. As pedras, frutas não ouro, árvores e posições de grama também são espalhados de forma aleatória na floresta a cada partida do jogo. As árvores, ao serem alocadas, devem ficar sobre posições onde há grama. Salvo este caso, dois elementos estáticos não podem ocupar a mesma posição na floresta. A posição inicial dos dois competidores também é aleatória.

Os tipos de frutas e de árvores, com as suas respectivas propriedades, são:

- maracujá (fruta ouro): um ponto de vitória;
- coco (agilidade): dobra o número de pontos de movimento de uma rodada;
- abacate (força): dobra a força do competidor;
- laranja (antídoto): anula o efeito do consumo de uma fruta bichada (mais sobre isso a seguir);

Frutas sem propriedades extras, o número de cada tipo é aleatório:

- acerola;
- amora;
- goiaba.

Haverá um total de 7 tipos distintos de frutas. Cada uma das frutas fornecem poder de força 1 ao competidor.

Durante sua coleta, uma fruta pode estar bichada, causando um efeito colateral que impossibilita o competidor de se mover na rodada seguinte. Se o competidor tiver uma fruta antídoto, pode consumi-la neste momento, para que sua próxima rodada não seja anulada. Qualquer tipo de fruta pode ser bichado, e as frutas com essa característica são sorteadas aleatoriamente no início da partida. O bicho da fruta é um elemento surpresa. Ou seja, o jogador só sabe se a fruta está bichada após catá-la. Caso tenha uma propriedade especial, a fruta não a perde caso esteja bichada.

A força do competidor deve ser atualizada à medida que a quantidade de frutas na mochila for alterada, seja por adicionar novas frutas ou por perder algumas em um conflito. O competidor pode consumir frutas em qualquer momento durante a sua vez de jogar.

Sobre a mochila do competidor:

- Possibilita o competidor carregar e usar as frutas em sua vez de jogar;
- Possui uma capacidade em número de frutas. Para simplificar, será considerado que cada fruta tem peso 1 (imagine que estamos lidando com uma porção da fruta em questão).

A configuração deve ser salva em um arquivo, que pode ser posteriormente carregado para visualização de um exemplo do terreno ou para iniciar um novo jogo. Ele deve obrigatoriamente ser um arquivo de texto e seguir **exatamente** o formato abaixo em relação às palavras usadas e sua ordem:

```
dimensao 6
pedras 7
maracuja 3 1
laranja 2 1
abacate 2 3
coco 2 1
acerola 1 2
amora 1 1
goiaba 1 1
bichadas 25
mochila 10
```

As declarações deste exemplo mostram que o terreno é de 6x6 (=36) posições, das quais 7 são ocupadas por pedra. O restante é ocupado automaticamente por grama, onde podemos colocar árvores.

A linha "maracuja 3 1" indica que o jogo terá 3 maracujás, dos quais 1 começa no chão. Os outros 2 aparecerão nas rodadas posteriores seguindo as regras da Seção 5.

Para cada fruta não ouro, os valores mudam um pouco de sentido. Eles indicam o número de árvores e a quantidade de frutas que vai estar no chão no início do jogo. Exemplo: a linha "laranja 2 1" indica que o terreno terá duas laranjeiras e começa com 1 laranja no chão.

A penúltima linha indica a porcentagem de chance de uma fruta ser gerada com bicho (neste caso, há 25% de chance de ser bichada), e a última linha a capacidade da mochila.

## 5. Vitória

Como já foi apresentado, há uma quantidade inicial de frutas no chão, e depois as árvores darão uma fruta quando o jogador se posicionar abaixo delas. A única exceção são as frutas ouro. Elas começam com uma certa quantidade pelo chão e não terão árvores para elas. Além disso, não deve ser possível comer os maracujás (azedo!) durante o jogo. Imaginemos que o banquete do vitorioso será composto por mousse e suco :)

Como o pé de maracujá é uma planta que cresce se apoiando em outras árvores (ela não é uma árvore), os maracujás que não estão no chão no início do jogo aparecerão na vizinhança 8 de duas árvores aleatórias a cada rodada par. Desta forma, em uma rodada par, aparecerá um maracujá no início da vez de um jogador e outro maracujá no início da vez do segundo. A ideia é que maracujás crescem apoiados nas árvores do jogo, e podem aparecer ao redor de qualquer uma delas. Depois que o número de frutas ouro é atingido, o jogo não as produz mais. Vence o jogo quem armazenar mais da metade dos maracujás em sua mochila primeiro. A verificação da vitória é sempre feita no início da vez do jogador, ou seja: o jogador vence oficialmente na rodada seguinte após o consumo dos maracujás necessários para vencer.

## 6. Interface

É requisito do trabalho a construção de uma interface gráfica para o jogo. A interface gráfica deve duas áreas principais:

- **Criação de terreno:** área do jogo onde o usuário fornece os valores configuráveis (vide seção anterior) e o jogo exibe um exemplo de terreno gerado com esses valores. O jogo deve validar se a entrada fornecida pelo usuário é compatível com o tamanho da floresta e posições disponíveis. Esta área deve oferecer a opção de salvar o terreno, onde um arquivo será criado com as configurações de jogo (vide exemplo na Seção 4). Aqui também é possível abrir um arquivo gerado e gerar um terreno de exemplo.
- **Iniciar jogo:** solicita ao usuário que insira os nomes dos jogadores e escolha um terreno, salvo em arquivo. Com esses dados, essa área deve iniciar o jogo e exibir o terreno visto de cima, atualizado de acordo com as jogadas dos usuários. Essa área também deve conter um painel de status, com uma narrativa das jogadas e orientações aos usuários.

É importante que a floresta e todos os elementos de um jogo sejam exibidos na interface. Além disso, é esperado que os eventos que ocorram sejam apresentados ao usuário de maneira intuitiva. Por exemplo: deve ser mostrado quem está se movimentando, para onde está indo, qual ação foi feita, qual o resultado da ação, etc.

O trabalho deverá ter uma classe principal.Principal.java que deve executar o programa e mostrar a interface.

## 7. Relatório

O relatório deve descrever de forma sucinta o que foi feito no trabalho e as soluções de implementação que vocês desenvolveram para os principais problemas encontrados.

## 8. Avaliação

A entrega do trabalho será feita em três etapas.

**Etapa 1:** modelagem das principais classes do sistema. Seguir notação UML ou padrão simplificado que será dado em aula.

A entrega deve ser um documento contendo:

- A descrição e representação das classes: descrição breve, seus atributos e tipos dos atributos, seus métodos com assinatura. Não é preciso implementar os métodos da classe nesta etapa mas é preciso explicar o que será feito em cada um deles.
- Relacionamentos entre classes: herança, associação, composição, agregação.

**Prazo de entrega:** 30/08/2024 04/09/2024

**Etapa 2:** projeto Java contendo a área **Criação do terreno**, com interface gráfica e javadoc.

A entrega deve ser um arquivo comprimido contendo: **relatório**, o **código fonte** e um **arquivo .jar** do projeto para execução do jogo.

O código fonte deve estar organizado em um projeto da IDE Eclipse 2024-06, e deve ser usado o JDK 22. A nota do grupo será zero se a aplicação não executar.

**Prazo de entrega:** 23/09/2024 03/10/2024

**Etapa 3:** projeto Java contendo o jogo final, com as áreas **Criação do terreno** e **Iniciar jogo**, com interface gráfica e javadoc.

A entrega deve ser um arquivo comprimido contendo: **relatório**, o **código fonte** e um **arquivo .jar** do projeto para execução do jogo.

O código fonte deve estar organizado em um projeto da IDE Eclipse 2024-06, e deve ser usado o JDK 22. A nota do grupo será zero se a aplicação não executar.

**Prazo de entrega:** 18/10/2024

Fator criatividade/capricho: vale até **um ponto extra** na nota final do trabalho.

As tabelas a seguir descrevem os critérios de avaliação em cada etapa.

<b>Etapa 1</b>		
<b>Funcionalidade</b>		<b>Porcentagem da nota da etapa</b>
Representação dos principais elementos do jogo e suas interações	Terreno	10
	Elementos estáticos	10
	Elementos dinâmicos	20
	Ações	10
	Funcionalidades do jogo	10
Organização do documento, clareza na apresentação das ideias e escrita		20
Uso adequado dos conceitos de programação orientada a objetos		20
<b>Total (nota vale 25% da nota da Unidade 2)</b>		<b>100</b>

<b>Etapa 2</b>		
<b>Funcionalidade</b>		<b>Porcentagem da nota da etapa</b>
Lógica do jogo	Representação do terreno em termos de EDs	10
	Validação dos parâmetros de entrada	10
	Criação do terreno de acordo com a sua configuração (número de elementos na cena)	20
	Visualização de um terreno a partir de arquivo	10
	Armazenamento do terreno criado em arquivo	5
Tratamento de erros		5
Interface e facilidade de uso		20
Organização do código fonte e javadoc		10
Relatório		10
<b>Total (nota vale 35% da nota da Unidade 2)</b>		<b>100</b>

<b>Etapa 3</b>		
<b>Funcionalidade</b>		<b>Porcentagem da nota da etapa</b>
Lógica do jogo	Carregamento do jogo a partir de arquivo	5
	Ações causadas pelas pedras e bordas do terreno	5
	Ações causadas pelas frutas	15
	Ações causadas pelas árvores	10
	Ações causadas pelo empurrão	10
	Movimento via teclado e mouse	10
Tratamento de erros		5
Interface e facilidade de uso		20
Organização do código fonte e javadoc		10
Relatório		10
<b>Total (nota vale 40% da nota da Unidade 2)</b>		<b>100</b>

## **9. Importante**

O trabalho prático deverá ser feito em grupos de duas a quatro pessoas, utilizando a linguagem Java. Grupos individuais serão aceitos apenas mediante justificativa. Os grupos devem se cadastrar na planilha:

<https://docs.google.com/spreadsheets/d/1Wz2AephtKI9fEMkrtNpsxtd-VFu93wbjNzqJDjsQOL0/edit?usp=sharing>

**Prazo de preenchimento da planilha: 20/08/2024.**

Mais algumas observações relevantes:

- A submissão das tarefas do projeto deve ser feita via SIGAA, nas datas previstas na Seção 8.
- Qualquer suspeita de plágio resultará em nota zero para todos os envolvidos.
- A depender do andamento da disciplina pode haver alterações no prazo.

**Bom trabalho!**