**1. Desenho da Grade (Tabuleiro)**

O tabuleiro será uma **grade bidimensional** (no exemplo abaixo, uma 5x5), onde cada célula é uma posição que o jogador e o monstro podem ocupar. Cada posição na grade será representada por **coordenadas** no formato (x, y), onde:

* x representa a linha.
* y representa a coluna.

Aqui está uma visualização conceitual de uma grade 5x5 com coordenadas:

0 1 2 3 4

+---+---+---+---+---+

0 |(0,0) (0,1) (0,2) (0,3) (0,4) |

+---+---+---+---+---+

1 |(1,0) (1,1) (1,2) (1,3) (1,4) |

+---+---+---+---+---+

2 |(2,0) (2,1) (2,2) (2,3) (2,4) |

+---+---+---+---+---+

3 |(3,0) (3,1) (3,2) (3,3) (3,4) |

+---+---+---+---+---+

4 |(4,0) (4,1) (4,2) (4,3) (4,4) |

+---+---+---+---+---+

* Cada célula representa uma posição possível no tabuleiro.
* O jogador e o monstro serão colocados aleatoriamente em uma dessas posições no início do jogo.
* O jogador tenta adivinhar a posição do monstro fornecendo coordenadas, por exemplo, (2,3).

**2. Movimentação do Jogador**

* O jogador **não se move** fisicamente no tabuleiro, mas faz **tentativas** de capturar o monstro fornecendo coordenadas.
* Em cada turno, o jogador dará coordenadas como um palpite, e o sistema dirá se o jogador acertou ou não.

**Exemplo de jogada do jogador:**

1. A posição atual do jogador não importa, mas ele faz uma tentativa escolhendo, por exemplo, a posição (2,3).
2. O jogo verifica se o monstro está na posição (2,3). Se estiver, o jogador vence. Caso contrário, o jogo continua e o monstro se move.

**3. Movimentação do Monstro**

* O monstro começa em uma posição aleatória na grade.
* Após cada tentativa errada do jogador, o monstro **se move para uma nova posição aleatória** dentro dos limites da grade.
* A movimentação do monstro será controlada pelo sistema, com base em números aleatórios gerados para x e y.

**Regras da movimentação do monstro:**

1. O monstro se move para uma nova posição **aleatória** após cada tentativa do jogador.
2. O monstro pode se mover para qualquer posição na grade, desde que não seja fora dos limites (ou seja, as coordenadas têm que estar entre 0 e 4 em uma grade 5x5).
3. Opcionalmente, o monstro pode se mover em uma direção "limitada" — como apenas mover-se uma célula por vez (ex.: de (2,3) para (3,3) ou (2,4)) em vez de saltar para qualquer parte da grade.

**4. Ciclo de Movimentação: Exemplo**

Vamos fazer um ciclo de movimentação para entender o comportamento do jogo.

**Passo 1: Inicialização**

* **Posição do jogador**: O jogador começa com nenhuma posição, pois apenas faz tentativas.
* **Posição inicial do monstro**: O monstro é colocado na posição (3,2) (oculto para o jogador).

**Passo 2: Tentativa do jogador**

* O jogador faz uma tentativa e escolhe as coordenadas (1,1).

**Passo 3: Verificação e movimento**

* O sistema verifica se o monstro está em (1,1). Como o monstro está em (3,2), a tentativa falha.
* O monstro se move para uma nova posição, por exemplo, (4,4) (decidida aleatoriamente).

**Passo 4: Nova tentativa do jogador**

* O jogador faz uma nova tentativa, escolhendo as coordenadas (3,2) (a antiga posição do monstro).
* O sistema verifica e informa que o monstro não está mais ali.
* O monstro se move novamente para uma nova posição, por exemplo, (0,0).

**O ciclo continua até que:**

* O jogador adivinhe a posição correta e capture o monstro, ou
* O jogador perca (se o monstro "escapar" ou o número de tentativas se esgotar).

**5. Possíveis Regras Adicionais**

* Podemos adicionar algumas **dicas de proximidade** para ajudar o jogador. Por exemplo, se o jogador errar por muito pouco, o sistema pode dizer algo como "Você está perto".
  + Isso pode ser feito calculando a **distância manhattan** (a soma das diferenças absolutas entre as coordenadas de duas posições). Quanto menor o valor, mais perto o jogador está.

**Exemplo de distância Manhattan:**

* O jogador escolhe (2,3) e o monstro está em (3,4).
* A distância seria abs(2-3) + abs(3-4) = 1 + 1 = 2. O sistema poderia dizer "Você está relativamente perto".

**6. Fluxo de Interações Resumido**

1. **Início do jogo**:
   * O tabuleiro (grade) é gerado.
   * O monstro é posicionado em uma célula aleatória, oculta ao jogador.
   * O jogador começa a fazer tentativas.
2. **Loop de jogo**:
   * O jogador dá coordenadas para tentar capturar o monstro.
   * O sistema verifica se o jogador acertou.
   * Se o jogador erra, o monstro se move aleatoriamente.
   * O loop continua até que o jogador capture o monstro ou perca.
3. **Condições de término**:
   * **Vitória**: O jogador captura o monstro ao adivinhar corretamente sua posição.
   * **Derrota**: O jogador perde se o monstro escapar ou se esgotar o número de tentativas.

**7. Opções Futuras:**

Esse desenho conceitual da grade e da movimentação é simples, mas pode ser expandido futuramente com:

* **Níveis de dificuldade**, ajustando o tamanho da grade (ex.: 7x7 para mais dificuldade).
* **Monstros diferentes** com padrões de movimentação específicos (por exemplo, movimentos mais previsíveis ou monstros que se movem para cantos).
* **Itens ou armadilhas** no tabuleiro, como obstáculos que dificultam o movimento.