

# Trabalho 1

## Métodos Formais

30 de setembro de 2024

## Contents

<b>1</b>	<b>Informações importantes</b>	<b>1</b>
<b>2</b>	<b>Uma batalha vencível</b>	<b>1</b>
2.1	Ataques . . . . .	2
2.2	Paralisia . . . . .	2
2.3	Cegueira ( <i>blind</i> ) . . . . .	2
2.4	Transformação selvagem ( <i>Wild Shape</i> ) . . . . .	2
2.5	Ilusão . . . . .	2
2.6	Criaturas . . . . .	2
2.6.1	Monstros . . . . .	2
2.6.2	Personagens . . . . .	3
2.7	Iniciativa . . . . .	3
<b>3</b>	<b>Atividades do trabalho</b>	<b>3</b>
3.1	Especificação com as regras do jogo . . . . .	3
3.2	Invariantes . . . . .	3
3.3	Especificação com estratégia . . . . .	4
3.4	Especificação com dois monstros . . . . .	4

## 1 Informações importantes

- Data de entrega: 30 de outubro (quarta-feira)
- Trabalho individual ou em dupla
- Especificações a serem entregues em TLA+ ou Quint.
- Formato de apresentação: Explicação breve do código (somente para a professora) + implementação de uma pequena modificação por membro.
- Nota: 50% código entregue + 50% apresentação.

## 2 Uma batalha vencível

De forma semelhante aos exemplos de jogo da velha e pokemons, você deve especificar um jogo onde as partes fazem jogadas aleatórias, mostrando que assim ambas as partes podem vencer; e então especificar uma versão desse jogo onde os personagens usam uma estratégia que garante que eles não percam.

As regras dessa batalha são inspiradas em jogo de RPG, porém um tanto exageradas para que possamos explorar algumas correspondências interessantes com problemas e soluções reais.

## 2.1 Ataques

Ataques são a forma de dar dano em criaturas. O dano é determinado por um inteiro, e ao receber um ataque, uma criatura perde uma quantidade de pontos de vida (HP - *Health Points*) igual ao dano do ataque. Uma criatura morre quando seus pontos de vida chegam a zero.

## 2.2 Paralisia

Criaturas podem ser paralisadas. Criaturas paralisadas não podem fazer nada em seu turno.

- A paralisia é permanente, e só pode ser removida se um personagem (não paralisado) usar seu turno para isso, ajudando um jogador paralisado e assim removendo sua paralisia.

## 2.3 Cegueira (*blind*)

Personagens da classe caçador tem habilidade de cegar um monstro. Ataques de um monstro cego causam 0 de dano. A cegueira dura um turno, de forma que o monstro deixa de estar cego ao terminar seu turno.

## 2.4 Transformação selvagem (*Wild Shape*)

Personagens da classe druida tem habilidade de se transformar em urso. Ao se transformar em urso, o druida passa a, temporariamente, ter 60 pontos de vida. Se o urso morrer, o druida volta a ter a quantidade de pontos de vida que tinha antes de se transformar. A transformação selvagem dura até o início do próximo turno do druida, onde ele pode escolher se transformar novamente ou fazer outra ação.

- Note que o urso nunca tem um turno. Quando chega a vez do druida, a transformação expira. Logo, ele não consegue fazer nenhuma ação como urso.

**Monstros tem medo de animais:** A ameaça (*threat*) que um urso causa nos monstros é maior, de forma que monstros sempre atacam o urso se ele estiver presente.

## 2.5 Ilusão

Personagens da classe mago tem habilidade de criar uma ilusão de uma ovelha com 1 ponto de vida. A ilusão não pode atacar, mas pode ser atacada. A ilusão dura até o início do próximo turno do mago, onde ele pode escolher recriar a ilusão ou fazer outra ação.

- A ilusão não tem um turno.

**Monstros tem medo de animais:** A ameaça (*threat*) que uma ilusão causa nos monstros é a mesma do urso, de forma que os monstros vão sempre preferir atacar ovelhas e ursos se esses estiverem presentes.

## 2.6 Criaturas

Essa batalha tem várias criaturas lutando entre si, entre elas monstros e personagens

- Monstros não atacam outros monstros
- Personagens não atacam outros personagens
- Podemos assumir que há somente um personagem de cada classe

### 2.6.1 Monstros

Todos os monstros são iguais. Monstros tem 100 pontos de vida iniciais, e podem fazer as seguintes ações em seus turnos (uma ação por turno)

1. Atacar um personagem Os ataques de monstro sempre dão 20 de dano, exceto se esse for o primeiro turno de **toda** a batalha (ninguém jogou ainda), onde o ataque é mais fraco (ele ainda está carregando todo o seu poder). No primeiro turno, um monstro dá apenas 10 de dano.

2. Paralizar um personagem Um monstro pode paralizar um personagem. Essa paralisia é permanente, e só é removida quando outro personagem usa sua ação para remover a paralisia de um jogador específico.

### 2.6.2 Personagens

Os ataques de personagem causam 10 de dano e tem 20 pontos de vida iniciais, independente da classe. Eles podem executar uma das seguintes ações em seu turno:

1. Mago
  - Atacar um monstro
  - Remover paralisia de um personagem
  - Criar uma ilusão
2. Caçador
  - Atacar um monstro
  - Remover paralisia de um personagem
  - Cegar um monstro
3. Druida
  - Atacar um monstro
  - Remover paralisia de um personagem
  - Se transformar em urso

## 2.7 Iniciativa

No início da batalha, cada criatura roda um d20 (dado com 20 faces, de 1 até 20) para determinar sua iniciativa. Aqueles com maior iniciativa jogam primeiro, e os com menor iniciativa jogam por último. Se duas ou mais criaturas tiverem a mesma iniciativa, a ordem que elas jogam entre si não importa. Nesse caso, fica a critério de vocês qual o comportamento exato. Se uma criatura A tem iniciativa maior que a criatura B, A deve jogar antes de B. Após todas as criaturas jogarem uma vez, o ciclo reinicia seguindo as mesmas regras e a mesma iniciativa.

## 3 Atividades do trabalho

### 3.1 Especificação com as regras do jogo

Primeiramente, escreva uma especificação descrevendo o que pode acontecer nesse jogo. Considere que exista um único monstro, e um personagem de cada classe. Nessa versão, deve ser possível tanto que o monstro quanto que algum dos jogadores morra.

- Inclua uma (ou mais) variável(is) que registre informações relevantes sobre o que aconteceu no último turno, assim como fizemos para o exemplo do pokemon.

### 3.2 Invariantes

Escreva duas invariantes:

1. O monstro não morre
2. Nenhum personagem morre

Nenhuma delas deve ser verdadeira nessa versão.

Você pode usar simuladores ao invés de *model checkers* para testar as invariantes em todas as etapas deste trabalho. Idealmente, devíamos usar model checkers, mas o modelo deste trabalho tem estados demais e execuções muito longas para uso de *model checkers*. Usaremos *model checkers* nos exemplos reais de sistemas distribuídos que veremos na disciplina.

### 3.3 Especificação com estratégia

Agora, modifique a especificação (mas salve o arquivo! Você precisa entregar as duas) para que os personagens utilizem uma estratégia, ao escolher suas ações, de forma que a invariante “nenhum personagem morre” seja verdadeira.

### 3.4 Especificação com dois monstros

Por último, modifique a especificação anterior (com estratégia) para que a batalha seja contra dois monstros. Você pode escolher entregar um arquivo a mais com essa versão, ou simplesmente entregar com essa parte comentada, já que pode ser uma modificação de apenas uma linha.

Essa batalha é impossível, mesmo com as estratégias empregadas. Assim, a invariante “nenhum personagem morre” deve ser violada. Se os personagens estiverem sobrevivendo na sua versão, pode ser que alguma regra não esteja definida corretamente.