

Especificar vs Implementar

Aula para disciplina de Métodos Formais

Gabriela Moreira

Departamento de Ciência da Computação - DCC
Universidade do Estado de Santa Catarina - UDESC

23 de setembro de 2024

Conteúdo

Exercício Pokemons

Especificar vs Implementar

Outline

Exercício Pokemons

Especificar vs Implementar

Exercício Pokemons

- Primeiros 20min de aula serão para vocês continuarem o exercício
 - Objetivo principal é que lembrem onde estavam, qual era a dificuldade, dúvidas, etc.
 - Se alguém terminou, pode me chamar para ter feedback da solução
- Depois, veremos uma sugestão de resolução.

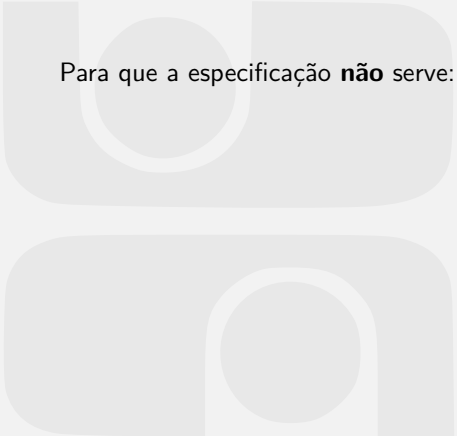
Outline

Exercício Pokemons

Especificar vs Implementar

Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:



Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:

- Para jogar pokemon

Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:

- Para jogar pokemon
 - Não temos input/output pros jogadores

Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:

- Para jogar pokemon
 - Não temos input/output pros jogadores
- Para analisar o balanceamento (i.e. Charmander ganha do Squirtle 40% das vezes)

Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:

- Para jogar pokemon
 - Não temos input/output pros jogadores
- Para analisar o balanceamento (i.e. Charmander ganha do Squirtle 40% das vezes)
 - Quint e TLA+ não fazem análise estatística

Por que estamos escrevendo uma especificação para um jogo de pokemon?

Para que a especificação **não** serve:

- Para jogar pokemon
 - Não temos input/output pros jogadores
- Para analisar o balanceamento (i.e. Charmander ganha do Squirtle 40% das vezes)
 - Quint e TLA+ não fazem análise estatística
 - Poderíamos checar que uma batalha é suficientemente balanceada (i.e. chance de vitória está entre 40 e 60%) com um model checker estatístico como o PRISM.

Por que estamos escrevendo uma especificação para um jogo de pokemon? II

Para que ela **serve**:

- Explicar pokemon pra quem não conhece pokemon

Por que estamos escrevendo uma especificação para um jogo de pokemon? II

Para que ela **serve**:

- Explicar pokemon pra quem não conhece pokemon
 - A especificação em si é uma descrição precisa do que pode acontecer e do que é esperado
 - Usando simulação ou *model checking*, podemos obter execuções exemplo que ajudam no entendimento

Por que estamos escrevendo uma especificação para um jogo de pokemon? II

Para que ela **serve**:

- Explicar pokemon pra quem não conhece pokemon
 - A especificação em si é uma descrição precisa do que pode acontecer e do que é esperado
 - Usando simulação ou *model checking*, podemos obter execuções exemplo que ajudam no entendimento
- Verificar propriedades

Por que estamos escrevendo uma especificação para um jogo de pokemon? II

Para que ela **serve**:

- Explicar pokemon pra quem não conhece pokemon
 - A especificação em si é uma descrição precisa do que pode acontecer e do que é esperado
 - Usando simulação ou *model checking*, podemos obter execuções exemplo que ajudam no entendimento
- Verificar propriedades
 - Verificar que algo desejável pode acontecer (i.e. é possível um pokemon ficar zero ou menos de vida)
 - Verificar que algo importante é verdade **sempre** (i.e. usando uma estratégia, o Squirtle sempre vence o Charmander)

Mas por que só não implementar e simular?

Por que só não fazemos isso em [insira linguagem de programação] e usamos um laço de repetição para testar N (ou todos) os cenários?

Mas por que só não implementar e simular?

Por que só não fazemos isso em [insira linguagem de programação] e usamos um laço de repetição para testar N (ou todos) os cenários?

Porque:

- Quint/TLA+ tem restrições para que seja possível fazer *model checking*
- O simulador é reutilizável, com toda a estratégia de simulação e interface
 - Blogpost do Andrew Helwer: TLA is more than a DSL for breadth-first search

Mas por que só não implementar e simular? II

Ah, mas se eu fizer em [insira linguagem de programação], eu já posso usar o programa para jogar também.

Mas por que só não implementar e simular? II

Ah, mas se eu fizer em [insira linguagem de programação], eu já posso usar o programa para jogar também.

- Para isso, seria necessário cuidar da distribuição/autorização dos jogadores
 - Jogador 1 só pode escolher os ataques do time A, e jogador 2 só pode escolher os ataques do time B
 - Isso diminui o nível de abstração, e pode tornar aquilo que é mais importante mais difícil de compreender.
 - No caso dos pokemons, isso é uma mudança pequena. Em alguns sistemas, a tarefa de buscar quais as possíveis escolhas de usuários (e outras formas de não-determinismo) no código é bem complexa.

Motivação para o trabalho



Motivação para o trabalho

- O principal motivo para o exercício de pokemons foi trabalhar em um contexto semelhante ao do nosso primeiro trabalho, que envolverá jogos de RPG de mesa.

Motivação para o trabalho

- O principal motivo para o exercício de pokemons foi trabalhar em um contexto semelhante ao do nosso primeiro trabalho, que envolverá jogos de RPG de mesa.
- Com algumas regras de RPG, conseguimos alguns problemas análogos aos problemas de sistemas distribuídos, campo onde métodos formais são melhor empregados.

Motivação para o trabalho

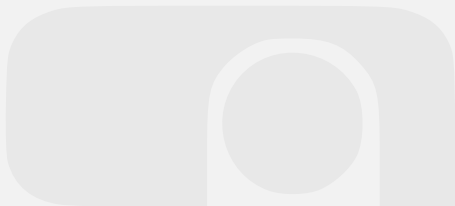
- O principal motivo para o exercício de pokemons foi trabalhar em um contexto semelhante ao do nosso primeiro trabalho, que envolverá jogos de RPG de mesa.
- Com algumas regras de RPG, conseguimos alguns problemas análogos aos problemas de sistemas distribuídos, campo onde métodos formais são melhor empregados.
 - Jogos em si não são um bom exemplo por conta da falta de impacto (risco = impacto x ameaça). Geralmente as empresas investem em usar métodos formais onde o impacto de possíveis problemas é muito alto.

Motivação para o trabalho

- O principal motivo para o exercício de pokemons foi trabalhar em um contexto semelhante ao do nosso primeiro trabalho, que envolverá jogos de RPG de mesa.
- Com algumas regras de RPG, conseguimos alguns problemas análogos aos problemas de sistemas distribuídos, campo onde métodos formais são melhor empregados.
 - Jogos em si não são um bom exemplo por conta da falta de impacto ($\text{risco} = \text{impacto} \times \text{ameaça}$). Geralmente as empresas investem em usar métodos formais onde o impacto de possíveis problemas é muito alto.
 - Depois de passar o trabalho, vou explicar um pouco essas analogias. Uma analogia que se aplica ao jogo da velha e ao pokemon: estratégias podem ser vistas como protocolos. Protocolos definem passos que devem acontecer em cada cenário com objetivo de garantir alguma coisa.

Protocolos

Exemplos de protocolos:



Protocolos

Exemplos de protocolos:

- Entregar a prova de cabeça pra baixo e pedir que todos desvirem no mesmo momento é uma forma de garantir que todos tem o mesmo tempo de prova

Protocolos

Exemplos de protocolos:

- Entregar a prova de cabeça pra baixo e pedir que todos desvirem no mesmo momento é uma forma de garantir que todos tem o mesmo tempo de prova
- Na decolagem/pouso de um avião, precisamos abrir as janelas e levantar as mesinhas

Protocolos

Exemplos de protocolos:

- Entregar a prova de cabeça pra baixo e pedir que todos desvirem no mesmo momento é uma forma de garantir que todos tem o mesmo tempo de prova
- Na decolagem/pouso de um avião, precisamos abrir as janelas e levantar as mesinhas
- Sistemas de trocas em jogos ou, semelhantemente, sistemas de pagamento estilo mercado pago

Protocolos

Exemplos de protocolos:

- Entregar a prova de cabeça pra baixo e pedir que todos desvirem no mesmo momento é uma forma de garantir que todos tem o mesmo tempo de prova
- Na decolagem/pouso de um avião, precisamos abrir as janelas e levantar as mesinhas
- Sistemas de trocas em jogos ou, semelhantemente, sistemas de pagamento estilo mercado pago

Podemos entender protocolos como estratégias para garantir ou impedir que algo aconteça.

Podemos entender estratégias de jogo como um protocolo que leva à vitória.

Protocolos

Exemplos de protocolos:

- Entregar a prova de cabeça pra baixo e pedir que todos desviem no mesmo momento é uma forma de garantir que todos tem o mesmo tempo de prova
- Na decolagem/pouso de um avião, precisamos abrir as janelas e levantar as mesinhas
- Sistemas de trocas em jogos ou, semelhantemente, sistemas de pagamento estilo mercado pago

Podemos entender protocolos como estratégias para garantir ou impedir que algo aconteça.

Podemos entender estratégias de jogo como um protocolo que leva à vitória.

Veremos mais sobre protocolos na próxima fase da disciplina

Outros exemplos de jogos

- Secret santa (amigo secreto)
- Rock paper scissors (pedra, papel, tesoura)
- Mafia/Werewolf (cidade dorme)

Link: <https://github.com/informalsystems/quint/tree/main/examples/games>

Especificar vs Implementar

Aula para disciplina de Métodos Formais

Gabriela Moreira

Departamento de Ciência da Computação - DCC
Universidade do Estado de Santa Catarina - UDESC

23 de setembro de 2024