Relatório de Sistematização: Matemática para Computação

Aluno: Luiz Filippe Neves Morisson

Desafio 1: reserva de quartos

Premissa: Somente quartos limpos e não ocupados podem ser reservados.

Predicados:

* - limpo(q): o quarto `q` está limpo
* - ocupado(q): o quarto `q` está ocupado
* - reserve(q): o quarto `q` pode ser reservado

Expressão:

**∀q (reserve(q) ↔limpo(q) ¬ocupado(q))**

Justificativa: Garante que apenas quartos aptos possam ser reservados, conforme os critérios estabelecidos.

Desafio 2: acesso para limpeza

Premissa: Funcionários só podem acessar quartos para limpeza se eles não estiverem ocupados.

Predicados:

* - funcionario(f): `f` é funcionário
* - podeLimpar(f, q): o funcionário `f` pode acessar o quarto `q`
* - ocupado(q): o quarto `q` está ocupado

Expressão:

**∀f ∀q ((funcionario(f) ∧ ¬ocupado(q)) → podeLimpar(f, q))**

Justificativa: Evita que funcionários acessem quartos com hóspedes presentes.

Desafio 3: acesso ao lounge vip

Premissa: Hóspedes VIP têm acesso ao lounge VIP se a estadia for superior a 3 dias.

Predicados:

* - vip(v): `h` é VIP
* - estadia(h, d): `v` está hospedado por `d` dias
* - acessoVIP(h): `v` tem acesso ao lounge VIP

Expressão:

**∀v ∀d ((vip(h) ∧ estadia(h, d) ∧ d > 3) → acessoVIP(h))**

**Justificativa:** Controla o acesso ao lounge com base na importância e duração da estadia.

Desafio 4: serviços de quarto

Premissa: Apenas quartos ocupados podem solicitar serviços, que devem ser registrados e faturados.

Predicados:

* - solicitaServico(h, q): `h` solicita serviço para `q`
* - ocupado(q): `q` está ocupado
* - RegistrarServico(h, q): serviço registrado e faturado

Expressão:

**∀h ∀q ((solicitaservico(h, q) ∧ ocupado(q)) → registrarserviço(h, q))**

Justificativa: Garante a eficiência e o correto faturamento do serviço de quarto.

Desafio 5: controle de estacionamento

Premissa: Apenas veículos de hóspedes ou funcionários registrados podem acessar o estacionamento, respeitando o número de vagas.

Predicados:

* - Veiculo(v): `v` é um veículo
* - Pertence(v, x): `v` pertence a `x` (hóspede ou funcionário)
* - Registrado(x): `x` está registrado
* - VagasDisponiveis(n): existem `n` vagas
* - PodeEstacionar(v): `v` pode acessar o estacionamento

Expressão:

**∀x ∀v ∀n ((pertence(v,x) ∧ registrado (x) ∧ vagasdisponiveis (n) ∧ n 0) → podeestacionar(v))**

Justificativa: Controla o acesso ao estacionamento, mantendo a organização e evitando lotação.