

### Teste Baseado em Especificação

#### Exercícios Geração e Implementação de Casos de Teste por Valor Limite (parte 2)

#### Objetivo

O objetivo deste exercício é definir e implementar, usando JUnit, um conjunto de casos de teste (com base nas técnicas baseadas em especificação por particionamento e valor limite) que permita aceitar ou rejeitar as classes entregues. O trabalho deve ser realizado em grupos com no máximo 4 integrantes.

#### Problema

Estamos interessados em um sistema de informação para o controle do estacionamento pago de centros comerciais cujo funcionamento se baseia em cancelas.

O exercício consiste na modelagem e implementação da classe responsável pelo cálculo do valor a ser pago pelo ticket de estacionamento.

Com base na data e horário de entrada e saída informados e se o automóvel é cliente VIP, o cálculo do valor a ser pago pelo estacionamento segue as seguintes regras:

- Cliente VIP tem 50% de desconto sobre o valor final da tarifa.
- 15 minutos de cortesia, ou seja, o valor a ser pago é zero, para todo tipo de cliente.
- Até 2 horas (inclusive) de permanência, o valor é fixo, sendo atualmente de R\$5,90.
- Acima de 2 horas e que não seja pernoite, o valor é incrementado de R\$2,50 a cada 1 hora.
- Pernoite (saída após as 8:00 do dia seguinte, sendo que o estacionamento é fechado às 2:00 da manhã), o valor é de R\$50,00 por pernoite.

#### Roteiro de Atividades

1. A partir das definições do problema:
  - a. Defina um conjunto de casos de teste unitários para a classe responsável pelo cálculo do valor da tarifa. Os casos devem obrigatoriamente serem gerados através das técnicas baseadas em especificação por particionamento e valor limite. Apresentar os casos de teste de forma tabular.
  - b. Implemente o conjunto de casos de teste definido usando o JUnit. Utilize o conceito de testes parametrizados para modularizar os testes.
2. Execute os testes unitários sobre a implementação realizada. Colete informações para a escrita de um relatório conforme instruções a seguir.

O relatório final deve ser um arquivo ZIP contendo:

1. Código-fonte de todas as classes e testes desenvolvidos.
2. O conjunto de casos de teste de forma tabular.
3. Para cada defeito encontrado:
  - a. A falha observada;
  - b. O nome do teste que a detectou;

- c. De que maneira foi corrigido.

## Data de Entrega

No Moodle até o dia 28/03/2024.

## Avaliação

Os seguintes itens serão avaliados:

1. Qualidade do projeto dos casos de teste;
  - a. Correta utilização das técnicas de geração de casos de teste;
  - b. Número suficiente de casos de teste com dados pertinentes;
2. Implementação dos casos de teste em JUnit;
  - a. Correta implementação dos casos de teste em classes driver de acordo como enunciado do trabalho;
3. Análise dos resultados dos testes.