

Relatório dos casos de teste do exercício 8 da lista de testes baseados em especificação

Nomes: Augusto Baldino, Vinicius Petersen

Estes números informam o tipo de caso esperado, porém devidos a erros na classe barca, alguns resultados não serão os esperados.

```
/*
 * 0 - Identificador de assento inválido
 * 1 - Assento ocupado
 * 2 - Assento bloqueado devido a distribuição de peso
 * 3 - Ok, assento atribuído ao passageiro.
 */
```

Neste caso, era esperado o número 0 (assento inválido), porém devido a falhas no código, ele está recebendo o valor 3 (OK, assento atribuído ao passageiro).

```
assertEquals(0, barca.ocupaLugar(assentoInformado:"F00A01"));
4j.AssertionFailedError: expected: [0] but was: [3] at com.Augusto.BarcaTest.testOcupaLugar...
```

Erros em cada caso:

0- Lugar inválido

```
@Test
public void testOcupaLugarInvalido() {
    assertEquals(0, barca.ocupaLugar(assentoInformado:"F00A01"));
    assertEquals(0, barca.ocupaLugar(assentoInformado:"F01A00"));
    assertEquals(0, barca.ocupaLugar(assentoInformado:"F61A01"));
    assertEquals(0, barca.ocupaLugar(assentoInformado:"F20A21"));
}
```

```
// Verifica se é um assento valido
if (Pattern.matches("[F][0-8]{2}[A][0-8]{2}", assentoInformado) == false){
    return 0;
}
```

ro de 0 a 8 com dois

O outro erro é que se o assento for 20 ou se a fila ou o assento for 0.

```
assertEquals(1, barca.ocupaLugar(assentoInformado:"F45A05"));
}
```

2- Assento bloqueado devido a distrição de peso

3- Ok, assento atribuído ao passageiro

```
//Este caso está correto, porém os outros casos comprometem o funcionamento do teste, como o caso  
// 1, que retorna o valor 3 ao invés do 1.
```

```
@Test
```

```
public void testOcupaLugarValido() {  
    assertEquals(3, barca.ocupaLugar(assentoInformado:"F21A01"));  
    assertEquals(3, barca.ocupaLugar(assentoInformado:"F50A01"));  
    assertEquals(3, barca.ocupaLugar(assentoInformado:"F55A01"));  
    assertEquals(3, barca.ocupaLugar(assentoInformado:"F60A01"));  
}
```