Técnica baseada na especificação

**Método**: int[] encomendaCombustivel (int quantidade, boolean emergencia)

Se emergencia = **TRUE**

A gasolina só é vendida se:

5% da quantidade subtraído do valor do tanque de aditivo, o restante do tanque deve ser >= 0

70% da quantidade subtraído do valor do tanque de gasolina, o restante do tanque deve ser >= 0

25% da quantidade subtraído dos valores dos tanques de álcool somados, o restante deve ser >= 0

Se emergencia = **FALSE**

A gasolina só é vendida se:

5% da quantidade subtraído do valor do tanque de aditivo, o restante do tanque deve ser >= 125

70% da quantidade subtraído do valor do tanque de gasolina, o restante do tanque deve ser >= 2500

25% da quantidade subtraído dos valores dos tanques de álcool somados, o restante deve ser >= 625

**Classes de equivalência:**

Vamos considerar que todos os tanques estejam cheios (aditivo = **500lts**; gasolina = **10000lts**; alcool1 = **1250lts**; alcool2 = **1250lts**) e que estamos somando os valores dos dois tanques de álcool.

|  |  |  |
| --- | --- | --- |
| Emergência | Quantidade | Resultado esperado |
| TRUE | 5000 | [250, 6500, 625, 625] |
| TRUE | 15000 | [-1, -1, -1] |
| FALSE | 1500 | [425, 8950, ?] |
| FALSE | 8250 | [-1, 0, -1] |
| FALSE | -1 | [-2] |

Explicar as condições para os métodos de abastecer a seguir

**Método**: int recebeAditivo (int quantidade)

|  |  |
| --- | --- |
| Quantidade | Resultado esperado |
|  |  |
|  |  |
|  | exceção |

**Método**: int recebeGasolina (int quantidade)

|  |  |
| --- | --- |
| Quantidade | Resultado esperado |
|  |  |
|  |  |
|  | exceção |

**Método**: int recebeAlcool (int quantidade)

|  |  |
| --- | --- |
| Quantidade | Resultado esperado |
|  |  |
|  |  |
|  | exceção |

Técnica do valor limite

**Método**: encomendaCombustivel (int quantidade, boolean emergencia)

**Matriz para valor limite:**

Vamos considerar que todos os tanques estejam pela metade (aditivo = **250lts**; gasolina = **5000lts**; alcool1 = **625lts**; alcool2 = **625lts**). Se estivermos tratando de **in-points** vamos presumir que o tanque da condição esteja cheio.

O único parâmetro avaliado na matriz é a quantidade, além disso estamos somando os valores dos dois tanques de álcool.

Se emergencia = **TRUE**

A gasolina só é vendida se:

t.aditivo >= 0 && t.gasolina >= 0 && t.alcool >= 0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Condition | Type | t1 | t2 | t3 | t4 | t5 | t6 |
| t.aditivo - (e\*0.05)  >= 0 | on |  |  |  |  |  |  |
| off |  |  |  |  |  |  |
| (tanque cheio) | in |  |  |  |  |  |  |
| t.gasolina - (e\*0.7)  >= 0 | on |  |  |  |  |  |  |
| off |  |  |  |  |  |  |
| (tanque cheio) | in |  |  |  |  |  |  |
| t.alcool - (e\*0.25)  >= 0 | on |  |  |  |  |  |  |
| off |  |  |  |  |  |  |
| (tanque cheio) | in |  |  |  |  |  |  |

T1: entrada {2500, FALSE}, saída {[125, 8250, 937, 937]} // ou 936

T2: entrada {2511, FALSE}, saída {[-1, 0, 0]}

T3: entrada {3572, FALSE}, saída {[]}

T4: entrada {3570, FALSE}, saída {[]}

T5: entrada {, FALSE}, saída {[]}

T6: entrada {, FALSE}, saída {[]}

Se emergencia = **FALSE**

A gasolina só é vendida se:

t.aditivo >=125 && t.gasolina >=2500 && t.alcool >= 625

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Condition | Type | t1 | t2 | t3 | t4 | t5 | t6 |
| t.aditivo - (e\*0.05)  >= 125 | on | 2500 |  |  |  |  |  |
| off |  | 2511 |  |  |  |  |
| (tanque cheio) | in |  |  |  |  |  |  |
| t.gasolina - (e\*0.7)  >= 2500 | on |  |  | 3572 |  |  |  |
| off |  |  |  | 3573 |  |  |
| (tanque cheio) | in |  |  |  |  |  |  |
| t.alcool - (e\*0.25)  >= 625 | on |  |  |  |  | 2500 |  |
| off |  |  |  |  |  | 2503 |
| (tanque cheio) | in |  |  |  |  |  |  |

T1: entrada {2500, FALSE}, saída {[125, 8250, 937, 937]} // ou 936

T2: entrada {2511, FALSE}, saída {[-1, 0, 0]}

T3: entrada {3572, FALSE}, saída {[321, 2500, 803, 803]} // ou 802

T4: entrada {3573, FALSE}, saída {[0, -1, 0]}

T5: entrada {, FALSE}, saída {[375, 8250, 313]} // tem que ver o 313

T6: entrada {, FALSE}, saída {[0, 0, -1]}

**Método**: int recebeAditivo (int quantidade)

...

**Método**: int recebeGasolina (int quantidade)

...

**Método**: int recebeAlcool (int quantidade)

...

Coisas pra escrever no relatório:

Em uma linguagem orientada a objetos, além de usarmos os parâmetros para explorar classes de equivalência, devemos considerar também o estado dos atributos do objeto.

Pensando nisso, resolvemos usar valores diferentes no estado dos tanques antes de rodar cada técnica de teste.

data= 3.5;

inteiro = (int)Math.round(data); //4

os valores da tabela são assim pq tamo usando esse metodo

como temos três condições no método de vender gasolina, o numero de combinações de teste pode crescer rapidamente, logo devemos usar a estratégia do domínio simplicado onde utilizamos uma matriz para facilitar a criação de casos de teste.

A fim de seguir a estratégia proposta decidimos variar os in-points nos diferentes testes(quando é true ou false na emergencia). Isso permite verificar se o programa fornece resultados corretos para vários in-points.