

Discente: Clarissa Alves Soares

Respostas Lista de Exercícios Unidade 2

PROBLEMA 1: Um aplicativo desenvolvido para o relógio da APPLE checa o risco de AVC com base na temperatura corporal (medida por um sensor em Celsius) e na frequência cardíaca (medida em quantidade de batimentos por minuto). O método abaixo representa a funcionalidade principal deste aplicativo.

```
public class AVCRisk {  
    /*  
    O método retorna o risco do usuário do relógio desenvolver um AVC:  
    - Retorna FALSE se: temperatura <= 39 || card <= 118  
    - Retorna TRUE se: temperatura > 39 && card > 118  
    Caso os sensores apresentem algum problema eles podem enviar um dado NEGATIVO este método  
    lança a exceção checada chamada IllegalArgumentException.  
    */  
    public boolean checkRisk (double temp, int card) throws IllegalArgumentException {  
        ...  
    }  
}
```

1. Como você pode particionar o domínio de entrada do método acima? Em outras palavras quais classes de equivalência podem ser definidas para este método?

Esse método pode ser particionado de duas formas, considerando cada parâmetro separadamente, em que é feito um partionamento baseado em interface, ou pode-se identificar as características que correspondem a funcionalidade, incluindo informações semânticas ao teste, ao fazer a partição baseada em funcionalidade.

Assim, temos as seguintes classes de equivalência:

Particionamento do Domínio de Entrada Baseado em Interface*

	Inválido	Válido	
	< 0	= 0	> 0
Double temp	C1 -0.1	C3 0.0	C5 0.1
Int card	C2 -1	C4 0	C6 1

*Apenas é considerado a relação do tipo do parâmetro com zero.

Particionamento do Domínio de Entrada Baseado em Funcionalidade*

	True	False
Risco de AVC (Argumentos Válidos)	C7 (39.1, 119)	C8 (39.0, 119) C9 (39.1, 118)
Valores Negativos (Argumentos Inválidos)	C10 (-0.1, -1)	C11 (0.0, 0)

2. Liste os casos de teste são necessários para cobrir todas as classes.

Casos de Testes para Método CheckRisk

Caso de Teste	Classes de Equivalência	VALORES DE ENTRADA	RESULTADO ESPERADO
CT1	C1	(-0.1 , 0)	InvalidArgumentException
CT2	C2	(0.0,-1)	InvalidArgumentException
CT3	C3, C4 e C11	(0.0 , 0)	False
CT4	C5 e C6	(0.1, 1)	False
CT5	C7	(39.1, 119)	True
CT6	C8	(39.0, 119)	False
CT7	C9	(39.1, 118)	False
CT8	C10	(-0.1, -1)	InvalidArgumentException

3. Os testes que você criou acima atendem ao critério de Análise de Valor Limite? Caso não atendam quais seriam os novos casos de testes que você precisaria criar para atender este critério e por quê?

Não, pois o valor limite mínimo fora da fronteira não foi considerado nos casos de teste anteriores como pode ser visto nos quadros abaixo.

	temp < 0.0	0.0 <= temp <=39.0	temp > 39.0
On	0.0	0.0, 39.0	39.0
In	-0,1	39.0	39.1
Out	0.1	-0.1, 39.1	38.9
Off	-0.1	-0.1,	39.1

	card < 0	0 <= card <=118	card > 118
On	0	0,118	118
In	-1	118	119
Out	1	-1,119	117
Off	-1	-1,119	119

Assim, o novo caso de teste deve ser acrescentado: CT9 – (38.9,117), False.

PROBLEMA 2: Considere o método abaixo que converte temperaturas de Celsius para Fahrenheit e vice versa. O código desta classe foi adicionado no SIGAA.

```
public class Temperature Transformer{
    public Temperature convert(Temperature temp) throws Exception{
        ...
    }
}
```

1. Como você pode particionar o domínio de entrada do método acima? Em outras palavras quais classes de equivalência podem ser definidas para este método?

Esse método pode ser particionado de duas formas, considerando o parâmetro Temperature, sem levar em consideração o contexto ao fazer um partionamento baseado em interface, ou pode-se identificar as características que correspondem a funcionalidade do método que converte temperaturas de Fahrenheit para Celsius, e vice-versa, ao considerar informações semânticas ao teste, ao fazer a partição baseada em funcionalidade.

Assim, temos as seguintes classes de equivalência:

Particionamento do Domínio de Entrada Baseado em Interface*

	Válido			Inválido
Parâmetro	temp < 0.0	temp = 0.0	temp > 0.0	Formatos
CelsiusTemperature	C1 (-0.1)	C2 (0.0)	C3 (0.1)	C4 ("string") C5 (null)
	temp < 32	temp = 32	temp > 32	Formatos
FahrenheitTemperature	C6 (31.9)	C7 (32)	C8 (32.1)	C9 ("string") C10 (null)

*Apenas é considerado a relação do tipo do parâmetro (celsius ou fahrenheit) com zero e formatos inválidos.

Particionamento do Domínio de Entrada Baseado em Funcionalidade*

	Cararterística		True	False
Válido	Zero <= temp<= Freeze	Celsius para Fahrenheit	C11 0	C19 0.1
		Fahrenheit para Celsius	C12 32	C20 32.1
	temp >= Boil	Celsius para Fahrenheit	C13 100	C21 99.9
		Fahrenheit para Celsius	C14 212	C22 211.9
	temp = Zero	Celsius para Fahrenheit	C15 -273.0	C23 -273.1
		Fahrenheit para Celsius	C16 -459.4	C24 -459.5
Inválido	temp < Zero	Celsius para Fahrenheit	C17 -273.1	C25 -272.9
		Fahrenheit para Celsius	C18 -459.5	C26 -459.3

2. Liste os casos de teste são necessários para cobrir todas as classes.

Casos de Testes para Método Convert(temp) - CelsiusTemperature

Caso de Teste	Classes de Equivalência	VALORES DE ENTRADA (Celsius)	RESULTADO ESPERADO (Fahrenheit)
CT01	C1	-0.1	31,82
CT02	C2 e C11	0.0	32
CT03	C3 e C19	0.1	32,18
CT04	C4	"string"	Não permite.
CT05	C5	Null	NullPointerException
CT06	C13	100	212
CT07	C15	-273	TemperatureException
CT08	C17 e C23	-273.1	TemperatureException
CT09	C21	99.9	211,82
CT10	C25	-272.9	-459,22

Casos de Testes para Método Convert(temp) - FahrenheitTemperature

Caso de Teste	Classes de Equivalência	VALORES DE ENTRADA (Fahrenheit)	RESULTADO ESPERADO (Celsius)
CT11	C6	31.9	-0,05555556
CT12	C7 e C12	32.0	0
CT13	C8 e C20	32.1	0,05555556
CT14	C9	"string"	Não permite.
CT15	C10	Null	NullPointerException
CT16	C14	212	100
CT17	C16	-459.4	TemperatureException
CT18	C18 e C24	-459.5	TemperatureException
CT19	C22	211.9	99,944444
CT20	C26	-459.3	-272,94444

3. Os testes que você criou acima atendem ao critério de Análise de Valor Limite? Caso não atendam quais seriam os novos casos de testes que você precisaria criar para atender este critério e por quê?

Não, pois o valor limite mínimo fora da fronteira superior não foi considerado nos casos de teste anteriores como pode ser visto nos quadros abaixo.

Análise Valor Limite CelsiusTemperature

	temp < -273	-273 <= temp < 0.0	0.0 <= temp < 100	Tem >=100
On	-273.0	-273.0, 0.0	0.0, 100.0	100.0
In	-273.1	-272.9,-0.1	0.1, 99.9	100.1
Out	-272.9	-273.1, 0.1	0.0, 100	99.9
Off	-273.1	-273.1, -0.1	-0.1, 99.9	99.9

Análise Valor Limite FahrenheitTemperature

	temp < -459.4	-459.4 <= temp < 32	32 <= temp < 212	Tem >= 212
On	-459.4	-459.4, 32	32, 212	212
In	-459.5	-459.3, 31.9	32.1 , 211.9	212.1
Out	-459.3	-459.5, 32.1	31.9, 212	211.9
Off	-459.5	-459.5, 31.9	31.9, 211.9	211.9

Assim, os novos casos de teste que devem ser acrescentados são:

Caso de Teste	Valores de Entrada	Resultado Esperado
CT21	100.1 C	212,18 F
CT22	212.1 F	100,05556 C