CLASSE JAVA ORIGINAL:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.DELACAO) {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.DELACAO) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

**return** PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

JUNIT ORIGINAL:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

*fail*("Not yet implemented");

}

}

OBS.: Não consegui rodar a JUNIT, pois a classe JulgamentoPrisioneiro está quebrada.

TESTE 2

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

**return** PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(15, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

1. Montei a JUNIT, mas ainda obtenho erro ao rodar o método calculaPena.
2. Criei a enum class Resposta e alguns erros sumiram.

Ainda tenho resposta negativa no teste.

TESTE 3

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(15, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

1. Comentei o return PENA\_CONDENACAO\_CUMPLICES, pois não pode ter dois returns dentro de uma mesma condição.
2. Ainda obtenho erro no teste da JUNIT no teste 03

TESTE 4

CLASSE JAVA

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(15, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

1. Antes de efetuar este teste, tive que colocar um return fora de todas as condições, pois uma função não pode ficar sem retorno.
2. O teste 04 foi executado com sucesso

TESTE 5

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.: Agora testaremos a seguinte condição:

SE respostaPrisioneiroA == Resposta.***DELACAO***)

SE respostaPrisioneiroB == Resposta.***DELACAO***

O return deverá ser PENA\_CONDENACAO\_MUTUA

1. Obtive erro pois o código do retorno não bateu com a regra de negócio, o código da regra é 5 e o programado é 15
2. O teste 05 foi executado com erro

TESTE 6

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.: Mudei os códigos das penas conforme regra de negócio e testaremos a condição novamente:

SE respostaPrisioneiroA == Resposta.***DELACAO***)

SE respostaPrisioneiroB == Resposta.***DELACAO***

O return deverá ser PENA\_CONDENACAO\_MUTUA

1. O teste 06 foi executado com sucesso

TESTE 7

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 10;

**private** **int** PENA\_CONDENACAO\_MUTUA = 15;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 11;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_INOCENCIA;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaDelacaoANegacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(10, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

Testaremos a condição com o método testPenaDelacaoANegacaoB:

SE respostaPrisioneiroA == Resposta.***DELACAO***)

SE respostaPrisioneiroB == Resposta.***NEGACAO***

O return deverá ser PENA\_CONDENACAO\_INDIVIDUAL

1. O teste 07 foi executado com erro, pois o retorno da condicional está errado

TESTE 8

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 0;

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaDelacaoANegacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(10, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

Testaremos a condição com o método testPenaDelacaoANegacaoB:

SE respostaPrisioneiroA == Resposta.***DELACAO***)

SE respostaPrisioneiroB == Resposta.***NEGACAO***

O return deverá ser PENA\_CONDENACAO\_INDIVIDUAL

1. O teste 08 foi executado com sucesso.

TESTE 9

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 0;

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaDelacaoANegacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(10, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaNegacaoADelacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***NEGACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(1, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}OBS.:

Testaremos a condição com o método testPenaDelacaoANegacaoB:

SE respostaPrisioneiroA == Resposta.***NEGACAO***)

SE respostaPrisioneiroB == Resposta.***DELACAO***

O return deverá ser PENA\_CONDENACAO\_CUMPLICES

1. O teste 09 foi executado com erro, pois o retorno está errado.

TESTE 10

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 0;

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaDelacaoANegacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(10, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaNegacaoADelacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***NEGACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(1, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.: Mudei os dois últimos retornos conforme a regra de negócio

Testaremos a condição com o método testPenaDelacaoANegacaoB:

SE respostaPrisioneiroA == Resposta.***NEGACAO***)

SE respostaPrisioneiroB == Resposta.***DELACAO***

O return deverá ser PENA\_CONDENACAO\_CUMPLICES

1. O teste 10 foi executado com sucesso.

TESTE 11

CLASSE JAVA:

**public** **class** JulgamentoPrisioneiro {

**private** **int** PENA\_INOCENCIA = 0;

**private** **int** PENA\_CONDENACAO\_MUTUA = 5;

**private** **int** PENA\_CONDENACAO\_INDIVIDUAL = 10;

**private** **int** PENA\_CONDENACAO\_CUMPLICES = 1;

**public** **int** calculaPena(Resposta respostaPrisioneiroA, Resposta respostaPrisioneiroB) {

**if** (respostaPrisioneiroA == Resposta.***DELACAO***) {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_MUTUA;

} **else** {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

}

} **else** {

**if** (respostaPrisioneiroB == Resposta.***DELACAO***) {

**return** PENA\_CONDENACAO\_INDIVIDUAL;

// return PENA\_CONDENACAO\_CUMPLICES;

}

}

**return** PENA\_INOCENCIA;

}

}

JUNIT:

**class** JulgamentoPrisioneiroTest {

@Test

**void** test() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(5, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaDelacaoANegacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***DELACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(10, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaNegacaoADelacaoB() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***NEGACAO***;

Resposta rb = Resposta.***DELACAO***;

*assertEquals*(1, julgamentoPrisioneiro.calculaPena(ra, rb));

}

@Test

**void** testPenaNegacaoANegacaoA() {

JulgamentoPrisioneiro julgamentoPrisioneiro = **new** JulgamentoPrisioneiro();

Resposta ra = Resposta.***NEGACAO***;

Resposta rb = Resposta.***NEGACAO***;

*assertEquals*(0, julgamentoPrisioneiro.calculaPena(ra, rb));

}

}

OBS.:

Testaremos a condição com o método testPenaNegacaoANegacaoB:

SE respostaPrisioneiroA == Resposta.***NEGACAO***)

SE respostaPrisioneiroB == Resposta.***NEGACAO***

O return deverá ser PENA\_INOCENCIA

1. O teste 11 foi executado com sucesso.