Bombe

// [const]

**int** getNumero();

// [const]

**int** getX();

// [const]

**int** getY();

// [const]

**int** getAmplitude();

**int** getCompteARebours();

**boolean** vaExploser();

// [invariants]

// 0 <= getCompteARebours(B) <= 10

// vaExploser(B) = (getCompteARebours(B) = 0)

// \pre 3 <= amplitude <= 11

// \post getNumero() = num

// \post getX() = x

// \post getY() = y

// \post getAmplitude() = amplitude

// \post getCompteARebours() = 10

**void** init(**int** num, **int** x, **int** y, **int** amplitude);

// \pre getCompteARebours() != 0

// \post getNumero() = getNumero()@pre

// \post getX() = getX()@pre

// \post getY() = getY()@pre

// \post getAmplitude() = getAmplitude()@pre

// \post getCompteARebours() = getCompteARebours()@pre - 1

**void** diminuerCompteARebours();

Classe de Test – BombeAbstractTest

PRECONDITIONS

1. Objectif 1 : init
   1. cas positif

condition initiale : aucune

opération : L1.1=init(1, 3, 5, 6)

oracle : pas de message d’erreur

* 1. cas négatif

condition initiale : aucune

opération : L1.1=init(1, 3, 5, 12)

oracle : mauvaise amplitude

1. Objectif 2 : diminuerCompteARebours

2.1 cas positif

Condition initiale : I2.1 = init(1, 3, 5, 6)

Opération : L2.1 = diminuerCompteARebours (I2.1)

Oracle : pas de message d’erreur

2.2 cas négatif

Condition initiale : I2.1 = init(1, 3, 5, 0)

Opération : getBloc(I2.2)

Oracle : compte à rebours nul

POST-CONDITION

1. Objectif 3 : init

Condition initiale : aucune

opération : L3.1 = init(1, 3, 5, 6)

oracle : invariants vrais et getNumero(L3.1) = 1 et getX(L3.1) = 3 et getY(L3.1) = 5 et getAmplitude(L3.1) = 6

1. Objectif 4 : diminuerCompteARebours

Condition initiale : I4.1 = init(1, 3, 5, 6)

Opération : L4.1 = diminuerCompteARebours(I4.1)

Oracle : invariants vrais et t getNumero(L4.1) = 1 et getX(L4.1) = 3 et getY(L4.1) = 5 et getAmplitude(L4.1) = 6

INVARIANTS

1. Objectif 5 : invariant 1 - 0 <= getCompteARebours(B) <= 10

Condition initiale : I5.1 diminuerCompteARebours(init(1, 3, 5, 6))

Opération : L5.1 = diminuerCompteARebours(I5.1)

Oracle : invariants vrais

1. Objectif 6 : invariant 2 - vaExploser(B) = (getCompteARebours(B) = 0)

6.1 cas A

Condition initiale : I6.1 = init(4, 3, 5, 1)

Opération : L6.1 = diminuerCompteARebours(I6.1)

Oracle : vaExploser = true

6.2 cas B

Condition initiale : I6.1 = init(4, 3, 5, 2)

Opération : L6.1 = diminuerCompteARebours(I6.1)

Oracle : vaExploser = false