Position

Kevin Coquart, Quentin Bunel

18 avril 2014

Contents

1 Position

1.1 service : Position

1.2 types :int, boolean

1.3 observators :

1.3.1

- x : [Position] $\rightarrow$ int

- y : [Position] $\rightarrow$ int

- z : [Position] $\rightarrow$ int

- dirG : [Position] $\rightarrow$ boolean

- equals : [Position] $\times$ Position $\rightarrow$ boolean

- equals : [Position] $\times$ int $\times$ int $\times$ int $\rightarrow$ boolean

- collision : [Position] × Position → boolean

1.4   Constructors :

1.4.1

- init : int × int × int × boolean → [Position]

1.5   Operators :

1.5.1

- setX : [Position] × int → [Position]

- setY : [Position] × int → [Position]

- setZ : [Position] × int → [Position]

- setDir : [Position] × boolean → [Position]

- set : [Position] × int × int × int → [Position]

- set : [Position] × Position → [Position]

1.6   Observations :

1.6.1   [invariant]

- equals(P, p2) =def (x(P) = x(p2) ∧ y(P) = y(p2) ∧ z(P) = z(p2))

- equals(P, i, j, k) =def (x(P) = i ∧ y(P) = j ∧ z(P) = k)

- collsion(P, p2) =def equals(P, p2) ∨ equals(P, x(p2) - 1, y(p2), z(p2))

- set(P, p2) =def set(P, x(p2), y(p2), z(p2))

1.6.2 [init]

- x(init(i, j, k, d)) = i

- y(init(i, j, k, d)) = j

- z(init(i, j, k, d)) = k

- dirG(init(i, j, k, d)) = d

1.6.3 [setX]

- x(setX(P, n)) = n

- y(setX(P, n)) = y(P)

- z(setX(P, n)) = z(P)

- dirG(setX(P, n)) = dirG(P)

1.6.4 [setY]

- x(setY(P, n)) = x(P)

- y(setY(P, n)) = n

- z(setY(P, n)) = z(P)

- dirG(setY(P, n)) = dirG(P)

1.6.5 [setZ]

- x(setZ(P, n)) = x(P)

- y(setZ(P, n)) = y(P)

- z(setZ(P, n)) = n

- dirG(setZ(P, n)) = dirG(P)

1.6.6 [setDir]

- x(setDir(P, n)) = x(P)

- y(setDir(P, n)) = y(P)

- z(setDir(P, n)) = z(P)

- dirG(setDir(P, n)) = n

1.6.7 [set]

- x(set(P, i, j, k)) = i

- y(set(P, i, j, k)) = j

- z(set(P, i, j, k)) = k

- dirG(set(P, i, j, k)) = dirG(P)