

MoteurJeu

Kevin Coquart, Quentin Bunel

18 avril 2014

Contents

1 MoteurJeu

1.1 service : MoteurJeu

1.2 use : GestionCombat

1.3 types : String, int, boolean, enum FIN {GAGNEE, PERDUE, NULLE}, enum COMMANDE {RIEN, GAUCHE, DROITE, HAUT, BAS, SAUTER, FRAPPE, JETER, RAMASSER}

1.4 observators :

1.4.1

- $\text{const maxPasJeu} : [\text{MoteurJeu}] \rightarrow \text{int}$
- $\text{pasJeuCourant} : [\text{MoteurJeu}] \rightarrow \text{int}$
- $\text{estFini} : [\text{MoteurJeu}] \rightarrow \text{boolean}$
- $\text{resultatFinal} : [\text{MoteurJeu}] \rightarrow \text{FIN}$

– pre resultatFinal(M) require estFini(M)

- combat : [MoteurJeu] → GestionCombat

1.5 Constructors :

1.5.1

- init : int × int × int × int → [MoteurJeu]

– pre init(largeur, hauteur, profondeur, maxPas) require largeur > 0 ∧ hauteur > 0 ∧ profondeur > 0 ∧ maxPas > 0

1.6 Operators :

1.6.1

- pasJeu : [MoteurJeu] × COMMANDE × COMMANDE → [MoteurJeu]

– pre pasJeu(M, comAlex, comRyan) require ¬estFini(M) ∧ comAlex ∈ COMMANDE ∧ comRyan ∈ COMMANDE

- pasJeu : [MoteurJeu] × String × COMMANDE → [MoteurJeu]

– pre pasJeu(M, nom, com) require ¬estFini(M) ∧ nom = ("Alex" ∨ "Ryan") ∧ com ∈ COMMANDE

1.7 Observations :

1.7.1 [invariants]

- 0 ≤ pasJeuCourant(M) ≤ maxPasJeu(M)

- $\text{estFini}(M) = \min (\text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Alex"))$
 $\wedge \text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Ryan"))$
 $\vee \text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Slick"))$
 $\vee \text{pasJeuCourant}(M) = \text{maxPasJeu}(M))$
- $\text{resultatFinal}(M) = \min$
 - GAGNEE si $\text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Slick"))$
 $\wedge \neg(\text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Alex"))$
 $\wedge \text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Ryan"))$
 - PERDUE si $\neg \text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Slick"))$
 $\wedge (\text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Alex"))$
 $\wedge \text{Personnage}::\text{estVaincu}(\text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{get}("Ryan"))$
 - NULLE sinon

1.7.2 [init]

- $\text{maxPasJeu}(\text{init}(l, h, p, m)) = m$
- $\text{pasJeuCourant}(\text{init}(l, h, p, m)) = 0$
- $\text{combat}(\text{init}(l, h, p, m)) = \text{GestionCombat}::\text{init}(l, h, p)$

1.7.3 [pasJeu]

- $\text{pasJeuCourant}(\text{pasJeu}(M, cA, cR)) = \text{pasJeuCourant}(M) + 1$
- $\text{combat}(\text{pasJeu}(M, cA, cR)) = \text{GestionCombat}::\text{gerer}(\text{combat}(M), \text{Map}<\text{String}, \text{COMMANDE}>)$
 - avec $\text{put}("Alex", cA) \wedge \text{put}("Ryan", cR) \wedge \forall i \in [0 \dots \text{GestionCombat}::\text{mPerso}(\text{combat}(M)).\text{size} - 1]$ $\text{put}(\text{name}, \text{randomCmd}()) \wedge \text{si } (\text{Math.random}() < 0.1) \text{ alors}$

put(randomName(), randomCmd())

- pasJeuCourant(pasJeu(M,n,c))=pasJeuCourant(M) +1
- combat(pasJeu(M,n,c))=GestionCombat::gerer(combat(M),Map<String,COMMANDE>)
 - avec put(n, c) $\wedge \forall i \in [0 \dots \text{GestionCombat::mPerso}(\text{combat}(M)).\text{size}()-2]$ put(name, randomCmd()) \wedge si (Math.random() < 0.1) alors
put(randomName(), randomCmd())