

Déplacement singe hunter.

(Curly)

déplacement

→ élémentaire

↳ direction : H, B, G, D

↳ largeur : 1 case

→ ~~vers~~ vers le pirate le plus proche

→ automatique → obligat° de déplacement

→ vitesse

Singe Hunter

→ position

Il

→ case : terre (T) ou mer (M)

→ taille 1 et 2

Singes

→ 0, 1, n

→ singe ; position ; type (orologique ou charnier)

Pirates

→ nb : 0, 1, n

→ pirates : position, état (vivant / Mort)

Trésor

→ position

→ visibilité

- ① déplacement HBDG d'1 case
- ② Obligation de déplacement (sauf impossible)
- ③ Vers le pirate le plus proche
- ④ Contact / interaction

→ déplacement nok = mer, terre + singe

→ " ok terre, T + pirate, T + trésor, T + rhin

Diapo 128

Analyse Partitionnelle

- Limiter le nombre de test à réaliser
- Chaque DT teste un comportement
- 2 classes d'équivalence ne peuvent pas avoir le même comportement.

Partir des variables d'entrées OUI des données à voir.

- 4 phases
- ① préoccupations → (quels sont les entrées, les comportements, les sorties)
 - ② Pour chaque préc. calcul de classes d'équivalence
 - ③ Choix d'un représentant de chaque classe d'équivalence.
 - ④ Produit cartésien des ensembles.

Diapo 130. 3 classes d'équivalence $\mathbb{R}^+, \mathbb{R}^-, \{0\}$
 Jeu de test $\rightarrow JT_1 = \{-2.5, 0, 3.4\}$

~~Good~~
 dans $\mathbb{Z} = a + ib$ on a

$a \backslash b$	\mathbb{R}^-	$\{0\}$	\mathbb{R}^+
\mathbb{R}^-			
$\{0\}$			
\mathbb{R}^+			

Diapo 132

Triangle

	Aigu	Obtus	Droit
Scalare	(6, 5, 3)	(5, 6, 10)	(3, 4, 5)
Isocèle	(6, 1, 6)	(7, 4, 4)	($\sqrt{2}$, 2, $\sqrt{2}$)
équil	(4, 4, 4)	\emptyset	\emptyset

⚠ il faut ajouter le cas de "non-triangle"

⚠ prendre les cas particuliers : triangle plat, le pt. n'est pas un triangle.

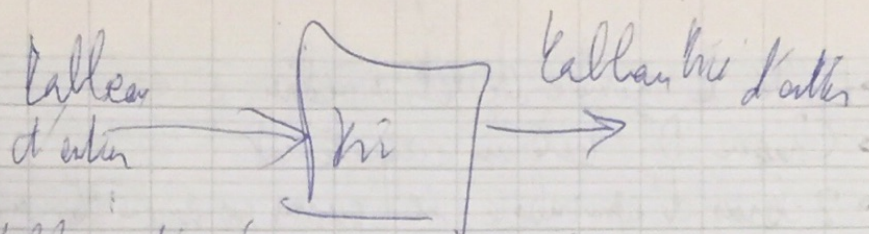


Tableau des entiers

① Taille $\rightarrow 0, 1, n$

② Éléments entiers $= < 0, = 0, > 0$

Tableau

→ Comportement $<, =, >$

redondance / non redondance

Tableau occ / non occ

→ et

Table \cap Table

①

Table \ occ	occ	! occ
0	[]	[]
1	[1]	X
n	$T_{occ}[n]$ $T_{!occ}[n]$	$T_{!occ}[n]$

2

entier	< 0	= 0	> 0
$T[1]$	[-35]	[0]	[2]
$T_{occ}[n]$	payement		

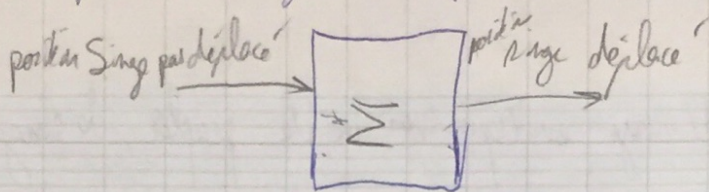
$T_{!occ}[n]$ sans redondance

$T_{occ}[n]$ avec "

$T_{occ}[n]$ avec redondance

$T_{!occ}[n]$ avec "

Déplacement singe erratique



Préoccupation :

- type de case d'arrivée terre / mer / \emptyset
- savoir si c'est abstrait :

A direction : H, B, G, D, impossible

• Occupation perso \rightarrow Singe / Pirate / \emptyset

• Occupation objet \rightarrow Rum / Trésor / Rum & Trésor / \emptyset

• Vitene

A

type case arrivé	terre	mer	\emptyset
Occupation			
Singe	(!D)	(!D)	(!D)
Pirate	(!D)	(!D)	(!D)
Perso	(!D)	(!D)	(!D)

A

Occupation objet	Pirate & Trésor	\emptyset & terre
Rum	(D)	(D)
Trésor	X	(D)
Rum & Trésor	(D)	(D)
\emptyset objet	(D)	(D)

déplacement d'objet erratique sur une île peuplée de singes et de pirates
déplacement:

→ élémentaire

↳ direction: H, B, G, D

↳ largeur: 1 case

→ aléatoire

→ automatique → obligé de déplacer

→ Vitesse

Singe erratique

→ position: coordonnées (type case)

Ilb

→ Case: terre (T) ou mer (M)

→ taille

Singes:

→ 0, 1, m

→ singe: position, type (erratique ou charism)

Pirates

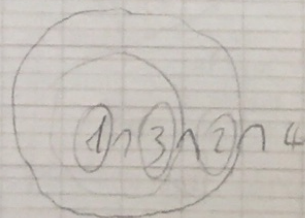
→ nb; 0, 1, m

→ pirate: position, état (vivant/mort)

Trésor

→ position

→ visibilité



test non régressif

faire 100 fois
pour déplacer
et 1000 pour
l'équilibre

① déplacement HBDG d'1 case

② Obligation de déplacement (sauf impossible) → Vérifier que si il est entouré de mer, il ne peut pas se déplacer

③ aléatoire

④ contexte/interaction

→ déplacement mort: mer, terre + singe

→ " ok: terre, T + pirate, T + trésor, T + rhum

Vérifier sur beaucoup de fois on a 25% pour chaque direction et ~~100~~ qu'on n'a pas 25% pour un petit échantillon (100)