

Référence Fourmissembleur

Format général

Un fichier `.brain` se présente sous la forme d'une suite de blocs d'instructions distingués par des labels de forme `<label>`. Il est possible d'ajouter des commentaires à un fichier en les préfixant par `;`.

Par exemple:

```
do_nothing:           ; Ne fait rien, quel beau commentaire par contre.  
    Drop
```

```
pick_up:  
    Pickup do_nothing ; On arrive ici.  
    Goto do_nothing   ; Oups, ça boucle.
```

Les instructions sont lues séquentiellement. Si un bloc ne se termine pas par une instruction de saut, la prochaine instruction sera celle du début du prochain bloc. Dans le cas où votre tout dernier bloc ne se termine pas par un saut (`Goto`, `Sense` ou `Roll`), vos méca-fourmis seront bloquées, faites bien attention !

Le Fourmissembleur n'est pas sensible à la case, `PickUp` et `pickup` sont équivalents pour le simulateur.

Les labels sont disjoints des instructions, nommer un bloc `move` est donc possible même si `move` peut aussi désigner une instruction.

Le simulateur est conçu pour renvoyer des erreurs (à peu près) claires en cas de problème dans votre `.brain`. Un check simple de syntaxe peut être fait via la commande `check`. Si un cas nous a échappé, n'hésitez pas à solliciter les organisateurs qui vous aideront à identifier la cause ou au moins vous soutiendront mentalement durant votre session de débogage.

Le code Fourmissembleur peut donner des fichiers très lourds si vous vous amusez à implémenter des états pour vos chères méca-fourmis. La commande `to-bin` du simulateur permet de réduire le poids de votre fichier en l'exportant vers un `.bbrain`.

Nous vous demanderons de bien vouloir rendre vos fichiers au format `.bbrain` pour les tournois. Aucune limite de taille n'est imposée, mais si votre fichier est trop gros pour être chargé, il sera considéré comme invalide. Plancez sur **8Go** maximum pour être certains de ne pas être embêtés.

Instructions

Mnémonique	Description	Repos
Sense <direction> <label_if> <label_else> <condition>	Saute à <code>label_if</code> si la condition est vérifiée dans la direction, sinon saute à <code>label_else</code>	1
Mark <index>	Marque le phéromone <code>index</code> de la case actuelle	1
Unmark <index>	Retire le phéromone <code>index</code> de la case actuelle	1
PickUp <label_error>	Ramasse une unité de nourriture sur la case actuelle, ou saute à	5

	label_error si aucune unité n'est ramassée	
Drop	Dépose une unité de nourriture sur la case actuelle si l'unité en porte une	5
TurnLeft	Tourne à gauche	1
TurnRight	Tourn à droite	1
Move <label_error>	Avance d'une case en avant. Si la case n'est pas accessible (déjà occupée, un rocher ou creusable), saute à label_error	20
MoveUp <label_error>	Remonte vers la surface. En cas d'échec (déjà à la surface, pas de trou creusé), saute à label_error	20
MoveDown <label_error>	Descends au sous-sol. En cas d'échec (déjà au sous-sol, pas de trou creusé), saute à label_error	20
Dig <label_error>	Creuse la cellule en face, ou saute à label_error en cas d'échec (rocher ou case vide)	25
Fill <label_error>	Rempli la cellule en face, ou saut à label_error en cas d'échec (rocher, pas en sous-col, cellule occupée, fourmillière)	25
DigUp <label_error>	Creuse un trou vers l'étage supérieur. En cas d'échec (déjà en surface, un rocher se trouve au dessus), saute à label_error. La case au dessus est automatiquement libérée si creusable	25
DigDown <label_error>	Creuse un trou vers l'étage inférieur. En cas d'échec (pas d'étage inférieur, un rocher se trouve en dessous), saute à label_error. La case en dessous est automatiquement libérée si creusable	25
FillUp	Rebouche un trou vers l'étage supérieur	25
FillDown	Rebouche un trou vers l'étage inférieur	25
Grab <label_error>	Attrape la méca-fourmi en face de l'unité. En cas d'échec (pas d'unité en face), saute à label_error	20 de stun + 30 de repos

Attack <label_error>	Attaque la méca-fourmi en face de l'unité. En cas d'échec (pas d'unité en face, ou unité qui n'est pas déjà attrapée), saute à label_error	30
Roll <n> <label_if> <label_else>	Lance un dé à n faces. Si le résultat est 0, va à label_if, sinon va à label_else	1
Goto <label>	Saut inconditionnel vers label	1

Tokens spécifiques

Directions

Mnémonique	Description
Here	La case courante de l'unité
Ahead	En face de l'unité
LeftAhead	À gauche de l'unité par rapport à son orientation actuelle
RightAhead	À droite de l'unité par rapport à son orientation actuelle
Above	Un étage au dessus de l'unité
Below	Un étage en dessous de l'unité

Conditions

Dans le cas où aucune case ne se trouve dans la direction donnée, on considère qu'un rocher la remplace.

Mnémonique	Description
Friend	Unité alliée
Enemy	Unité ennemie
Grabbed	Unité attrapée
FriendWithFood	Unité alliée portant de la nourriture
EnemyWithFood	Unité ennemie portant de la nourriture
Food	Contient au moins une unité de nourriture au sol
Rock	Est un rocher
Empty	Case vide, sans garantie sur le fait que de la nourriture s'y trouve ou non
Underground	La méca-fourmi est sous terre
Surface	La méca-fourmi est à la surface
HoleAbove	La case permet de remonter à l'étage supérieur
HoleBelow	La case permet de descendre à l'étage en dessous
Marker <index>	Le phéromone index est marqué
EnemyMarker	Au moins un phéromone ennemi est marqué
Home	Fourmillière alliée
EnemyHome	Fourmillière ennemie