

Ce challenge est basé sur un système de leagues.

Pour ce challenge, plusieurs ligues pour le même jeu seront disponibles. Quand vous aurez prouvé votre valeur contre le premier Boss, vous accéderez à la ligue supérieure et débloquerez de nouveaux adversaires.

Objectif

Contrôler plus de **cases** que votre adversaire à la fin du match.

Règles

Les **Robots** sont déployés dans un champ de débris électroniques abandonnées, leur fonction est de reconditionner des parcelles de s pour en faire de nouveaux objets.

Les robots sont aussi capables de se désassembler et de se répliquer, mais ils ont besoin de **ressources brutes** provenant de bâtiments appelés **recycleurs** que peuvent construire les robots.

Les bâtiments vont **recycler** toutes les cases autour d'eux en ressources brutes, pour faire simple ils convertissent les parcelles de débris en parcelles d'**herbe**, qui se trouvent en dessous.

Les joueurs contrôlent une **équipe** de ces robots au cours d'une **compétition amicale**, pour déterminer quelle équipe peut contrôler le plus de parcelles dans un champ de débris donné. Les équipes **marquent** des cases avec la couleur de leur équipe, avec les contraintes suivantes:

- Si des robots des deux équipes se retrouvent sur la même case, ils doivent se désassembler eux-même **un pour un**. Les robots sont donc enlevés du jeu, laissant ainsi au plus une équipe sur la case.
- Les robots **ne peuvent pas traverser l'herbe**, ceux qui sont encore sur une case quand elle est entièrement recyclée doivent donc eux-aussi se désassembler.

Une fois la partie terminée, les robots se réassembleront et retourneront travailler comme si de rien n'était.



Un robot de l'équipe bleue.



Un robot de l'équipe rouge.

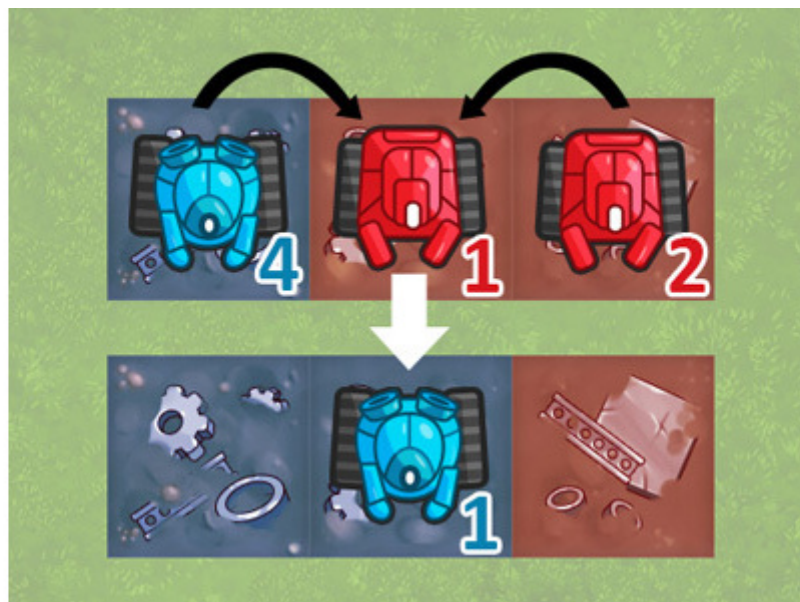
La carte

La partie est jouée sur une grille de taille variable. Chaque case sur la grille représente une parcelle de débris électroniques. Le but du jeu est de contrôler plus de cases que son adversaire, en les faisant **marquer** par des robots. Chaque case a les propriétés suivantes:

- **scrapAmount** : la quantité de matériau utilisable sur la case. Elle est égale au nombre de tours nécessaires pour être complètement recyclée. Si égale à 0, cette case est de l'**herbe**.
- **owner** : quel joueur contrôle cette case. Vaut **-1** si la case est neutre ou est de l'**herbe**

Les robots

N'importe quel nombre de robots peut occuper une case, mais si des unités des deux équipes finissent leur tour sur la même case, ils sont retirés du jeu **1 pour 1**. Après cela, si la case possède encore des robots, ces derniers marqueront la case.



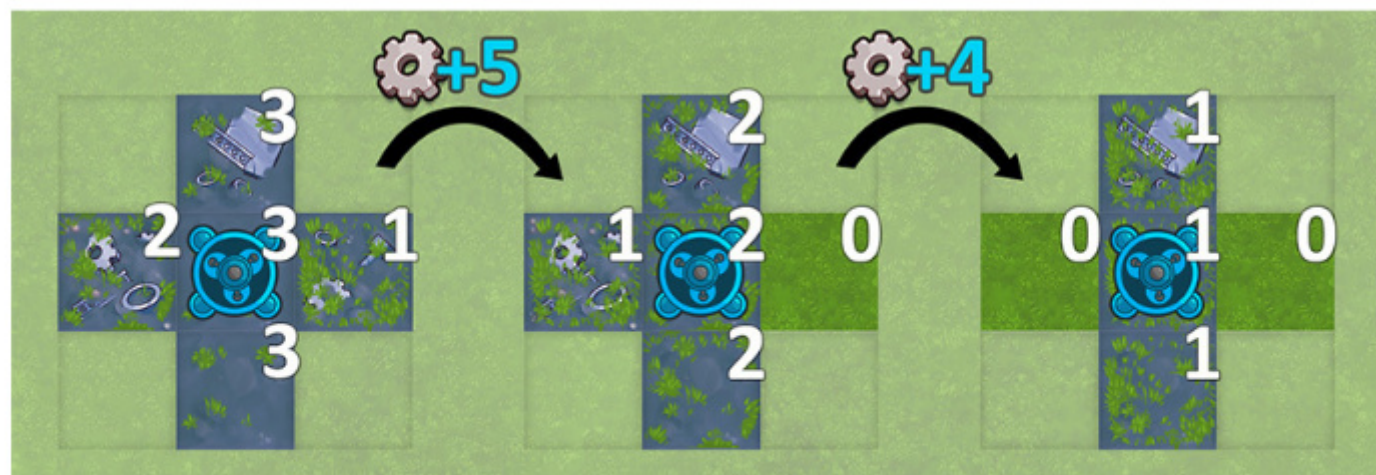
Après avoir bougé tous les robots sur la case du milieu, seul un robot bleu y reste et la tuile est marquée.

Les robots ne peuvent pas occuper une case d'**herbe** ou partager une case avec un **recycleur**

Les recycleurs

Les recycleurs sont des bâtiments qui occupent une case. Chaque tour, sa case et toutes les cases adjacentes sont utilisées pour être recyclées, réduisant ainsi leur **scrapAmount** et rapportant **1** unité de matériaux au propriétaire du recycleur.

Si la case sur laquelle est posée un recycleur est à court de **scrapAmount**, le recycleur est démonté.



Toute tuile à la portée de vos recycleurs vous rapporte **1 matériau** par tour et perd **1 scrapAmount**.

Une tuile donnée ne peut être recyclée qu'une seule fois par tour. Cela implique que son **scrapAmount** sera réduit de **1** même si les deux joueurs ont plusieurs recycleurs adjacents, leur rapportant chacun uniquement **1 matériau**.

Le Matériau

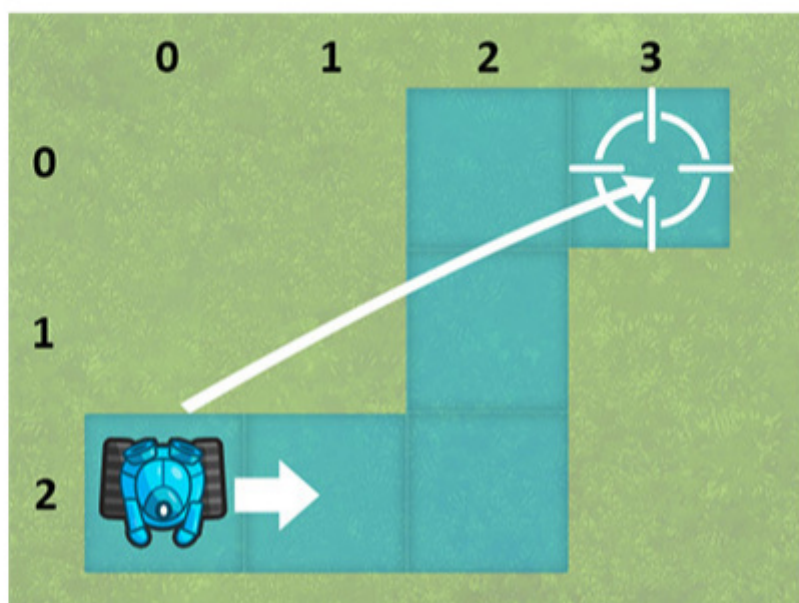
10 matériaux peuvent être dépensées pour créer un nouveau robot, ou pour construire un autre **recycleur**.

A la fin de chaque tour, les deux joueurs reçoivent **10** matériaux en bonus.

Les actions

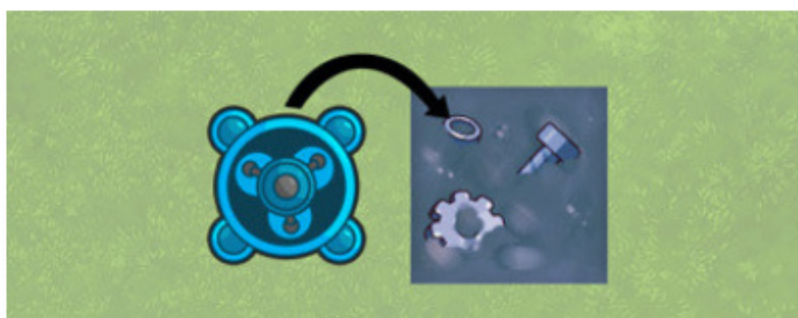
À chaque tour les joueurs peuvent entreprendre n'importe quel nombre d'actions valides, qui sont:

- **MOVE** : déplace un nombre donné d'unités d'une case à une case adjacente. Si la case ciblée n'est pas adjacente, les unités avanceront sur le chemin le plus court jusqu'à la destination.

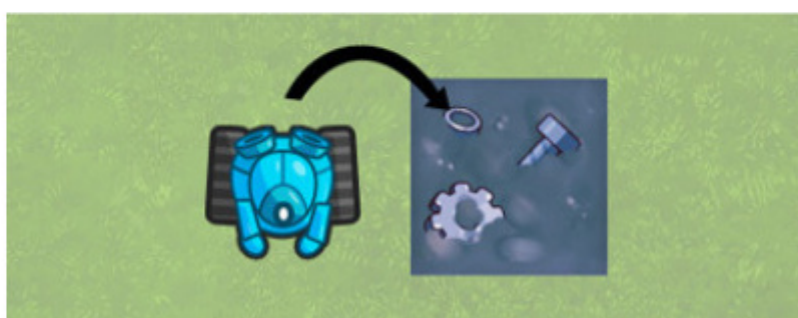


Un **MOVE** vers (3,0) fera marcher ce robots vers (1,2).

- **BUILD** : construit un recycleur sur une case vide spécifiée que le joueur contrôle.



- **SPAWN** : construit un nombre donné de robots sur la case spécifiée que le joueur contrôle.



L'ordre des actions sur un tour

1. Les actions **BUILD** sont complétées.
2. Les actions **MOVE** et **SPAWN** sont simultanément complétées. Un même robot ne peut faire ces deux actions pendant le même tour
3. Les unités des équipes opposées sur une case sont retirées **un pour un**.
4. Les robots restants marqueront les cases sur lesquelles elles se trouvent, changeant le **owner** de la case.
5. Les recycleurs recyclent la case sur lesquelles elles se trouvent et les 4 cases adjacentes tant qu'elles ne sont pas déjà de l'**herbe**
6. Les tuiles avec un **scrapAmount** à 0 sont maintenant de l'**herbe**. Les recycleurs ou les robots sur cette case sont retirés.

7. Les joueurs reçoivent de base **10** matériaux ainsi que ceux acquis lors du recyclage.

Conditions de Victoire

Le gagnant est celui qui contrôle le plus de **cases** après soit:

- Un joueur ne contrôle plus une seule case.
- **20** tours se sont passés sans qu'une seule case n'ait changé de **scrapAmount** ou d'**owner**.
- **200** tours ont été joués.

Conditions de Défaite

Votre programme ne répond pas dans le temps imparti ou il fournit une commande non reconnue.

Conseils de débogage

- Survolez une case pour voir plus d'informations sur celle-ci, y compris son **historique**.
- Utilisez la commande **MESSAGE** pour afficher du texte sur votre côté du HUD.
- Cliquez sur la roue dentée pour afficher les options visuels supplémentaires
- Utilisez le clavier pour contrôler l'action : espace pour play / pause, les flèches pour avancer pas à pas

Détails Techniques

- L'**owner** d'une case ne changera pas s'il y n'y a pas de robots dessus à la fin du tour.
- Si la cible d'une commande **MOVE** n'est **pas atteignable**, les robots iront sur la case la plus proche de la destination donnée, préférant celle la plus proche du centre de la carte.
- En choisissant un chemin pour la commande **MOVE** dont la cible est **éloignée** mais atteignable, les robots prendront le plus court chemin, préférant rester proche du centre de la carte si possible.
- Les commandes **MOVE** et **SPAWN** se produisent simultanément et ne peuvent pas entrer en conflit l'une avec l'autre. Cependant elles peuvent être invalidées par une commande **BUILD**, même si elle arrive plus tard dans la liste de commandes du joueur, ou fait partie des actions de l'adversaire.

Protocole de Jeu

Entrées d'Initialisation

Une ligne: deux entiers **width** et **height** pour la largeur et la hauteur de la carte. La case en haut à gauche est $(x,y) = (0,0)$.

Entrées pour Un Tour de Jeu

Première ligne: deux entiers **myMatter** et **oppMatter** pour la quantité de matériaux de chaque joueur.

Les **height** * **width** lignes suivantes: une ligne par case, commençant par celle en **(0,0)** et allant de gauche à droite, de haut en bas. Chaque case est représentée par **7** entiers:

Les **4** premières variables décrivent les propriétés pour cette case :

- **scrapAmount** : le nombre de fois que cette case peut être recyclée avant de devenir de l'**herbe**.
- **owner** :
 - **1** si vous contrôlez cette case.
 - **0** si votre adversaire contrôle cette case.
 - **-1** sinon.
- **units** : le nombre d'unités sur cette case. Ces unités appartiennent à l'**owner** de la case.
- **recycler** : **1** si il y a un recycleur sur cette case. Ce recycleur appartient à l'**owner** de cette case. **0** s'il n'y pas de recycleur sur cette case.

Les **3** variables suivantes sont là pour vous aider:

- **canBuild** : **1** si vous avez le droit de **BUILD** (construire) un recycleur sur cette case ce tour. **0** sinon.
- **canSpawn** : **1** si vous avez le droit de **SPAWN** (ajouter) des robots sur cette case ce tour. **0** sinon.
- **inRangeOfRecycler** : **1** si cette tuile se fera réduire sa **scrapAmount** par un recycleur à la fin du tour. **0** sinon.

Sortie

Toutes vos actions sur une ligne, séparées par un **:**

- **MOVE** **amount** **fromX** **fromY** **toX** **toY**. Pathfinding automatique.
- **BUILD** **x** **y**. Construit un recycleur sur une case libre contrôlée.
- **SPAWN** **amount** **x** **y**. Ajoute des robots sur une case contrôlée.
- **WAIT**. Ne fait rien.
- **MESSAGE** **text**. Affiche un texte sur votre côté du HUD.

Contraintes

$$12 \leq \text{width} \leq 15$$

$$6 \leq \text{height} \leq 7$$

Temps de réponse par tour \leq **50 ms**

Temps de réponse pour le premier tour \leq **1000 ms**

Qu'est-ce qui m'attend dans les ligues suivantes ?

- Des cartes plus grandes seront disponibles.


Keep Off The Grass!

La vie d'un **Recycloro-bot** est simple.

Noter l'emplacement de la ferraille, construire des recycleurs, aller dans le prochain champ de débris et répétez, tout en respectant la Première Loi: "**Ne Roulez Pas sur l'Herbe**". Mais parfois, même le plus joyeux des **Recycloro-bot** peut être un peu lassé par ces tâches répétitives.

C'est pourquoi, de temps en temps, ils aiment organiser **La Grande Ferraillaton**, une joute amicale entre deux équipes où celle qui marque le plus de débris avec sa couleur à la fin du temps imparti est déclarée vainqueur.

Cependant, pendant un match les robots ne peuvent utiliser que les ressources recyclées du tas de ferraille sur lequel ils se trouvent ! Tous les coups sont permis, même recycler à un tel point que la précieuse **herbe** est déblayée, coupant l'accès à votre adversaire... ou coupant la branche sur laquelle vous êtes assis, si vous ne faites pas attention...



Kit de Démarrage

Des IAs de base sont disponibles dans le [démarrage](#). Elles peuvent vous aider à appréhender votre propre IA.

Code Dource

Le code source de ce jeu est disponible [ici](#).