

# Itération 3 : Bilan

Bernard Julien, Robineau Thomas, Rouyer Hugolin, Russo Nicolas  
S5 RA-IL1

# Sommaire

- I. Répartition des tâches
- II. Détails
- III. Prochaine itération

# Répartition des tâches

- Amélioration de l'agent artificiel actuel (Julien)
- Début de conception de l'agent décentralisée (Hugolin et Nicolas)
- Tests et débogage (Hugolin)
- Automatisation de la recherche d'éléments (Nicolas)
- Système de temps (Thomas)
- Interface graphique (Thomas)

# Détails - Interface Graphique

- Affichage des recettes disponibles dans le niveau



Salade



Tomate cuit(e) et coupé(e)

- Ajout d'un en-tête affichant les différentes informations de la partie

● Joueur 0 Points : 0 Temps restant : 55 s ● Joueur 1

Fin de la partie, votre score est de : 10 points

Pour rappel, on a associé un entier à un état.

Cet entier (+1) correspond également au nombre de points attribués à chaque plat.



Salade



Tomate cuit(e) et coupé(e)

Salade : 1 pt

Tomate cuite et coupée : 4 pts

# Détails - Automatisation de la recherche d'éléments

- Pour l'agent artificiel, **recherche d'éléments** précis sur la **carte** (Cuisson, Découpe, Tomate, ...)
- Avant, boucle **switch** pour **chaque type** d'élément
- Désormais, mise en place de **String** pour chaque élément, et **comparaison** en fonction.
- Passer de **150 lignes** à **40 lignes**

```
case "SaladeTomate", "TomateSalade":  
    for (int i = 0; i < hauteur; i++) {  
        for (int j = 0; j < longueur; j++) {  
            if (objetsFixes[i][j] instanceof  
                PlanDeTravail planDeTravail)  
                if (planDeTravail.getInventaire().contains("Tomate") ||  
                    planDeTravail.getInventaire().contains("Salade"))  
                    coordonneesElem.add(new Point(i, j));  
            if (objetsDeplacables[i][j] != null)  
                if (objetsDeplacables[i][j].getNom().contains("Tomate") ||  
                    objetsDeplacables[i][j].getNom().contains("Salade"))  
                    coordonneesElem.add(new Point(i, j));  
        }  
    }  
    return coordonneesElem;
```

# IA: Objectif

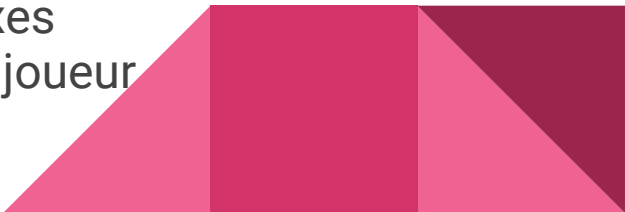
A long terme:

- Agent artificiel capable de prendre des décisions sur des niveaux complexes en collaborant.

A court terme:

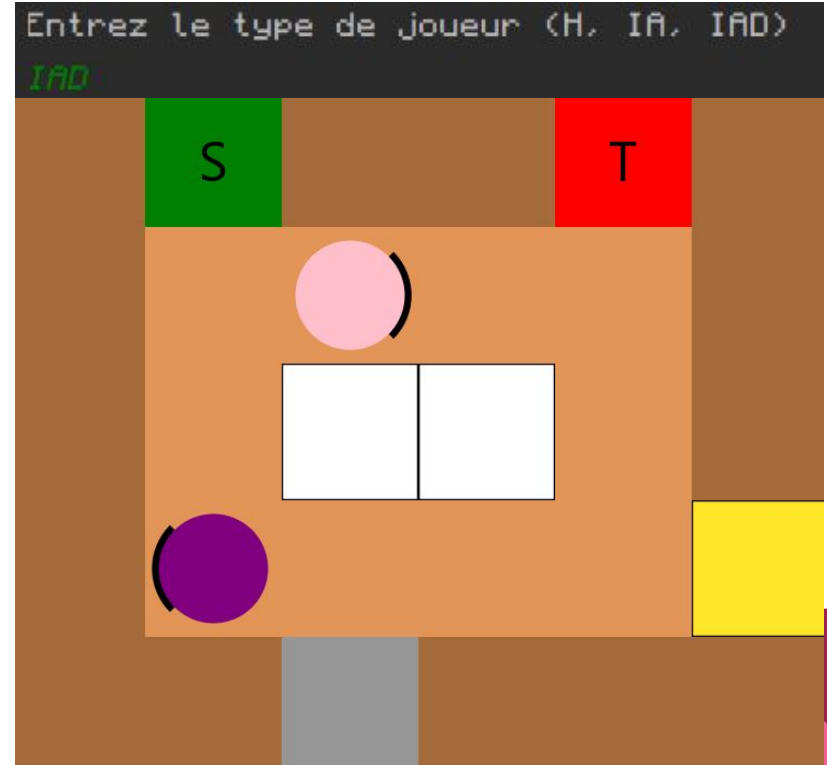
- Agent artificiel capable de réaliser des tâches relativement simples.
- Inversion du fonctionnement de l'agent:  $A^* \rightarrow \text{BFS} \Rightarrow \text{BFS} \rightarrow A^*$

A venir:

- Agent artificiel capable de réaliser des tâches complexes
  - Agent artificiel capable de prendre en compte l'autre joueur
- 

# Détails - Début de conception de l'IA décentralisée v1

- **Adaptation** de l'agent actuel
- Prise en compte des **autres joueurs**
  - Calcul du **chemin optimal** pour les autres joueurs;
  - Donne le **premier aliment** qu'ils vont créer;
  - On **retire** l'aliment en question dans la **liste des recettes** de l'agent.
- **Pas encore optimal**
  - N'arrive **pas au bout** de la recette
  - **Temps** de calcul assez **long**



# Mise au point sur le projet

- Mise en place d'un moteur de jeu pour récupérer la carte et les recettes ;
- Création d'une interface graphique simple et complète ;
- Système de recette pour les plats (avec états) ;
- Système de plan de travail, de transformateur, de générateur ;
- Mise en place d'un agent artificiel unique ;
  - Peut récupérer des aliments, les transformer et les déposer ;
  - La collaboration n'est pas encore implémentée ;
- Ajout de contrôles pour l'IA et le joueur.





# Prochaine itération

- Amélioration des performances générales de l'IA
- Amélioration de l'IA décentralisée

## Éventuellement

- Interface graphique améliorée (menu)
- Ajout de recettes plus complexes/nouveaux plats

