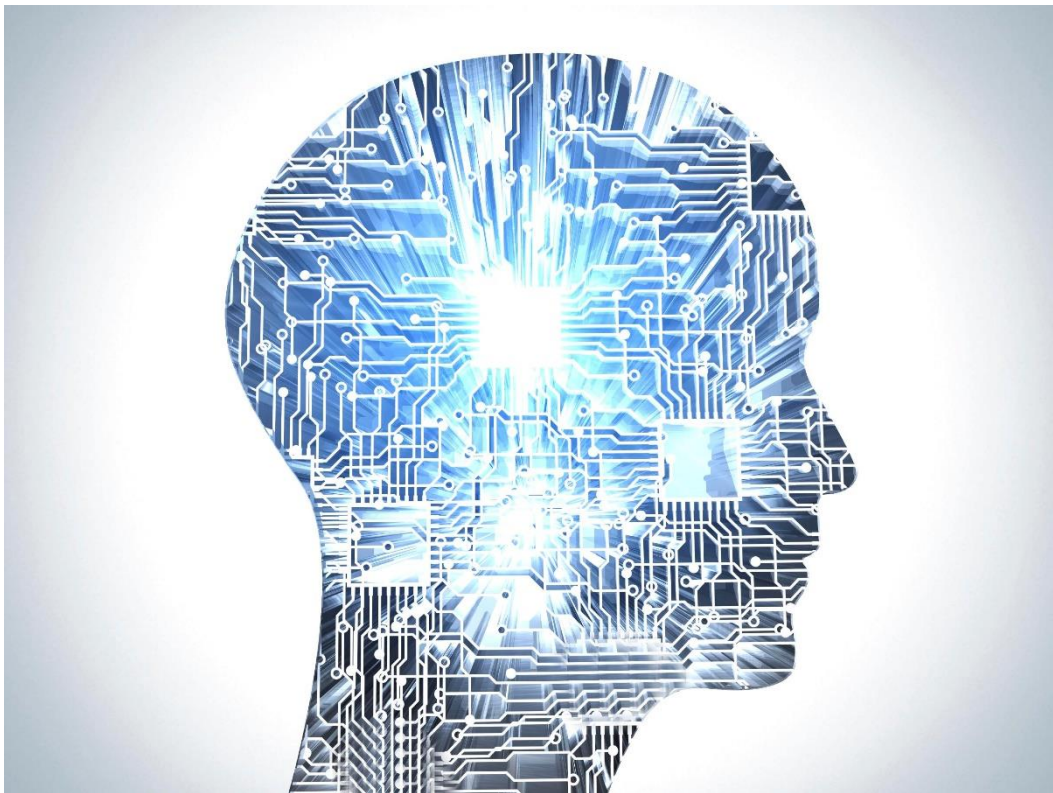


COMPTE RENDU :

**TP2 : UNIFICATION ET
RESOLUTION LOGIQUE**



Hammouda Hajer & Chaabene Amal

RT4 / Groupe 2

Introduction :

Le but de ce TP est d'implémenter l'algorithme d'unification.

Explication du travail réalisé :

Nous avons créé des fichiers qui vont contenir les atomes à unifier. Chaque fichier contient une ligne qui est composée de deux atomes.

Dans le code nous avons défini une classe « **unification** » qui contient les fonctions suivantes :

- **Fonction « extractFonction »** : cette fonction prend en argument un atome. Elle va identifier les fonctions parmi les éléments de cet atome et les mettre dans une liste et elle retourne cette dernière.
- **Fonction « extract1element »** : elle prend en argument un atome, elle fait appel à la fonction « extractFonction » si elle trouve un '(' pour extraire la fonction et la retourner comme un 1^{er} élément à traiter sinon elle prend l'élément avant la virgule qui peut être (constante/ variable). Elle retourne le 1^{er} élément.
- **Fonction « unifierAtome »** : elle prend en argument les deux atomes à unifier. Elle va identifier quel élément sera remplacé selon les types des éléments (constante/ variable/fonction).
nous avons les cas suivants :
 - Constante – constante (différents) : retourne « échec »
 - Constante – constante (égales)
 - Variable – variable
 - Variable – constante
 - Variable – fonction (si la variable existe dans fonction) : retourne « échec »
 - Variable – fonction (variable n'existe pas dans la fonction)
 - Autre : retourne « échec »
- **Fonction « substitution »** : cette fonction prend en argument les deux atomes, elle va remplacer l'atome1 par l'atome2, et ajouter cette modification à la chaîne « ch » qui va contenir la trace de chaque substitution.
(l'argument 1 est substitué par l'argument 2)
- **Fonction « unifier »** : cette fonction prend en argument deux atomes et à chaque fois on teste s'ils existent ou non. S'ils existent, on fait appel à la fonction « extract1element » pour chaque atome (p1 et p2) et on teste si les 2 éléments existent puis on teste si les deux éléments sont des fonctions on fait appel à la

fonction « unifier » il s'agit une récursivité, sinon on fait appel à la fonction « unifierAtome » pour les deux éléments (p1 et p2), on fait appel par suite à la fonction « substitution » pour faire la substitution et enfin on fait appel à la fonction « unifier » au reste des éléments. (à chaque fois on élimine les éléments traités)

L'utilisateur peut à chaque fois choisir le fichier et demander de sauvegarder la trace dans un fichier.

On fait appel au constructeur de la classe « unification » et on appelle la fonction « unifier » en passant en argument l'atome1 et 2.