Intervalles d'Allen

L'objectif du TME est d'implémenter en python le calcul des relations obtenues par composition de relations, ainsi que la création d'un graphe temporel grâce à l'algorithme de propagation d'Allen.

Les relations sont représentées par des chaînes de caractères.

Les fichiers Python, dûment commentés, sont à envoyer pour le compte rendu.

Exercice 1 Implémentation de la composition

1. Télécharger depuis le moodle le fichier `LRC_TME8_definitions_Allen.py` qui définit :
   - un dictionnaire `transpose` de type `dict[str: str]`, tel que `transpose[r]` fournit la transposition de la relation `r`.
   - un dictionnaire `symetrie` de type `dict[str: str]`, tel que `symetrie[r]` fournit la symétrie de la relation `r`.
   - un dictionnaire `compositionBase` de type `dict[tuple[str,str]: set[str]]`, dont les clés sont des couples de relations et tel que `compositionBase[(r1,r2)]` est l'ensemble des relations obtenues lorsqu'on compose la relation `r1` avec la relation `r2`. Ce dictionnaire représente la table de composition réduite donnée en cours (diapositive 35 du Cours 8).
2. Définir les fonctions `transposeSet` et `symetrieSet` telles que `transposeSet(S)` (resp. `symetrieSet(S)`) renvoie l'ensemble des relations transposées (resp. symétriques) des relations qui apparaissent dans `S`.
3. Définir la fonction `compose` telle que `compose(r1, r2)` renvoie l'ensemble des relations que l'on peut obtenir par composition des relations `r1` et `r2`.

   Il s'agit donc d'utiliser :
   - la table de composition directement
   - la transposition : $r_1 \circ r_2 = (r_2^t \circ r_1^t)^t$
   - la symétrie : $r_1 \circ r_2 = (r_1^s \circ r_2^s)^s$
   - les deux transformations simultanément : $r_1 \circ r_2 = (r_2^{st} \circ r_1^{st})^{ts}$

   Pour tester l'implémentation, vérifier par exemple que
   - $= \circ$ d $= \{$d$\}$
   - m $\circ$ d $= \{$d, o, s$\}$
   - ot $\circ > = \{>\}$
   - $> \circ$ e $= \{>\}$
   - ot $\circ$ m $= \{$dt,et,o$\}$
4. Définir la fonction `compositionSet` telle que `compositionSet(S1, S2)` renvoie l'ensemble des relations que l'on peut obtenir par composition des relations qui apparaissent dans l'ensemble `S1` avec les relations qui apparaissent dans l'ensemble `S2`.

   Il s'agit de généraliser la fonction précédente à des ensembles.

Exercice 2 Gestion du graphe temporel

L'implémentation de l'algorithme d'Allen peut ne pas suivre les indications de cet exercice si une autre solution vous paraît plus facile. Il est toutefois conseillé de représenter un graphe par :

- l'ensemble de ses nœuds
- l'ensemble des relations entre ses nœuds, par exemple à l'aide d'un dictionnaire ayant pour clés des couples de nœuds, et pour valeurs les ensembles des relations correspondantes.

Un graphe avec deux nœuds A et B et A{o,e}B sera donc représenté par :

- ses nœuds {'A','B'}
- ses relations {('A','B') :{'o','e'} }

1. Créer une classe `Graphe` ayant pour attributs `noeuds` et `relations`, ainsi qu'une fonction `getRelations` telle que `g.getRelations(i,j)` renvoie l'ensemble des relations entre les nœuds `i` et `j` du graphe `g`.

   Attention au traitement des cas où c'est la relation transposée qui est stockée dans le graphe ou s'il n'existe pas de relation stockée entre les nœuds `i` et `j`.
2. Créer une fonction `propagation` qui prend en entrée un graphe et deux nœuds de ce graphe, et utilise l'algorithme d'Allen pour propager la relation entre ces deux nœuds dans le reste du graphe. On suppose ici que les deux nœuds existent déjà, on ne fait que propager la relation en question.
3. Créer une fonction `ajouter` qui prend en entrée un graphe et une relation entre deux nœuds, et ajoute cette relation au graphe. On peut avoir les deux nœuds qui appartiennent déjà au graphe, l'un d'eux uniquement, ou aucun.

   Pour construire un graphe cohérent, il suffit maintenant de créer un graphe vide puis d'ajouter les relations une par une.
4. Tester la création de graphe avec des cas simples de graphes à trois nœuds A, B, C :
   - un graphe avec les relations A$\{<\}$B et A$\{>\}$C
   - un graphe avec les relations A$\{<\}$B et A$\{<\}$C
   - dans les deux cas, propager la relation B$\{=\}$C
5. Créer une fonction `retirer` qui prend en entrée un graphe et un nœud et retire du graphe ce nœud et toutes les relations qui le concernent (sans autre mise à jour).
6. Faire des versions 'verbose' de vos fonctions d'ajout et de propagation permettant d'afficher proprement toutes les étapes de la mise à jour.
7. On reprend l'exercice 1 de la feuille de TD 8 pour faire automatiquement toutes les propagations impliquées dans le raisonnement. Partir d'un graphe temporel vide et ajouter les relations au fur et à mesure qu'elle sont déduites :
   - Durant la première phase (Question 2) :
     — Ajouter la contrainte $t_M\{d\}t_C$ (obtenue par skolemisation de (b))
     — Ajouter la contrainte $t_S\{m\}t_C$ (obtenue par skolemisation de (d))
     — Vérifier que la propagation de cette dernière contrainte permet bien de déduire la relation $t_S\{<\}t_M$
   - Pour la question 3 :
     — Ajouter les contraintes $t_S\{<\}t_E$ et $t_E\{<\}t_M$ (obtenues par skolemisation via $T'' = t_E$ de la conclusion de la question précédente : résolution de (e) avec $T = t_S$ et $T' = t_M$, les prémisses de (e) ayant été prouvées par le raisonnement)
     — Vérifier que ces propagations permettent bien de déduire la relation $t_E\{d\}t_C$.
   - Pour la fin du raisonnement
     — Retirer le nœud t_S du graphe temporel.
     — Ajouter la contrainte $t_E\{o, d^t, e^t\}t_B$ (obtenue par skolemisation de (f) pour $T_E = t_E$).
     — Ajouter la contrainte $t_B\{<, m, m^t, >\}t_C$ (obtenue par résolution de (g) avec $OCCURS(Bruit, t_B)$ qui a été dérivée précédemment) et montrer que cela aboutit à une contradiction temporelle.
     — Refaire le même raisonnement sans retirer le nœud t_S pour constater qu'on obtient bien la même contradiction (mais avec des propagations plus longues).

   On a ainsi obtenu (en utilisant les versions 'verbose' de l'algorithme) une correction propre des parties du raisonnement de l'exercice 1 liées aux propagations dans le graphe temporel.