Ces structures de données implémentent la détection des réductions généralisées. Voir: Jouvelot, P., et Dehbonei, Babak, "A Unified Semantic Approach for the Vectorization and Parallelization of Generalized Reductions", ACM-ICS'89, Crete.

1 Evaluation Symbolique

Import expression from "ri.newgen"

Import entity from "ri.newgen"

Définition des expression symboliques utilisées pour la détection des réductions.

gexpression = `guard:expression x expression`

Une expression guardée est formée d'une garde et d'une expression symbolique. La garde indique quand l'expression symbolique est valide

sexpression = `gexpressions:gexpression*`

Une expression symbolique est formée d'une liste d'expressions gardées. Toutes les gardes sont mutuellement exclusives.

2 Pattern-matching

Definition des patterns qui décrivent les réductions généralisées.

pattern = `variable:entity x condition:expression x parameter:expression x operator:entity x indices:entity*`

Un `pattern` *décrit une réduction. La* `variable` *est l'entité sur laquelle la réduction est effectuée. La* `condition` *décrit la garde associée (dans le cas de conditionnelle, e.g., max). Le* `parameter` *définit la partie indépendante de la réduction. L'*`operator` *définit l'opération entre la variable et le paramètre. Les* `indices` *est la liste des variables d'unification qui doivent être égales à l'index de la boucle la plus englobante.*

3 Reduced loop

reduced_loops = `ordering:int*`

This is the list of loops that are parallel with a reduction.