GALLET Thomas
PONTANIER Baptiste
LEGLAND Jordan
THOMAS Jacques

4A

Documentation passe 2
Octobre 2017

I-Liste des erreurs lancées par nos classes de vérification de la syntaxe …

Pour le fichier ErreurContext: Ce fichier s'occupe de signaler des erreurs de contexte détectées par nos classes de verification, à savoir Verif et ReglesTypages.

Voici la **liste des exceptions lancées par ErreurContex**t :

- **ErreurAffectationInvalide** : Cette erreur est lancée lorsque le programme ***AffectCompatible*** détecte une incompatibilité de type lors d'une affectation (en suivant les règles définies dans context.txt). Elle renvoie **les types des deux membres de l'affectation ainsi que le numéro de ligne associé à l'erreur.**

- **ErreurOperationInvalide :** Cette erreur est lancée lorsque le programme ***AffectCompatible*** détecte une incompatibilité entre les deux opérandes dans le cas d'OpérationBinaire, ou alors une incompatibilité entre l'opérande et l'opéation dans le cas d'OpérationUnaire (Ex : Not b avec b un réel n'a pas de sens).Elle renvoie **les types du/des membres de l'opération ainsi que le numéro de ligne associé à l'erreur** .

- **ErreurIdentificateurNonDeclare**: Cette erreur est lancée lorsque le programme ***Verif*** détecte un identificateur dans la partie instructions qui n'a pas été déclaré dans la partie déclaration du programme. Elle renvoie **le nom de l'identificateur non déclaré ainsi que le numéro de ligne associé à l'erreur**.

- **ErreurIdentificateurDejaDeclare** : Cette erreur est lancée lorsque le programme ***Verif*** détecte un identifcateur qui est redéclaré dans la partie déclaration du programme.Elle renvoie **le nom de l'identificateur déjà déclaré ainsi que le numéro de ligne associé à la redéclaration**.

- **ErreurTypeInconnu**:Cette erreur est lancée lorsque le programme ***Verif*** détecte un mauvais typage de variable dans la zone de déclaration du programme (Ex : b : double).Elle renvoie **le type inconnu ainsi que le numéro de ligne associé à l'erreur**.

- **ErreurTypeRead** : Cette erreur est lancée lorsque le programme ***Verif*** détecte que l'argument de la fonction read n'est pas un integer ou un réel. Elle renvoie **les types supportés par la fonction Read ainsi que le numéro de ligne associé à l'erreur**.

- **ErreurTypeWrite** : Cette erreur est lancée lorsque le programme ***Verif*** détecte que l'argument de la fonction write n'est pas de type integer, réel, ou string. Elle renvoie **les types supportés par la fonction Write ainsi que le numéro de ligne associé à l'erreur**.

- **ErreurTypeWhile** : Cette erreur est lancée lorsque le programme ***Verif*** détecte que la condition exprimée dans le while n'est pas de type Boolean. Elle indique que **le type boolean est attendu et renvoie le numéro de ligne associé à l'erreur**.

- **ErreurTypeIf** : Cette erreur est lancée lorsque le programme ***Verif*** détecte que la condition exprimée dans le if n'est pas de type Boolean. Elle indique que **le type boolean est attendu et renvoie le numéro de ligne associé à l'erreur**.

- **ErreurTypeForVariable** : Cette erreur est lancée lorsque le programme ***ReglesTypage*** détecte que la variable de contrôle de la boucle for n'est pas de type integer. Elle indique **le type de la variable de contrôle et le type demandé et affiche également le numero de ligne associé à l'erreur**.

- ***ErreurTypeForBorne*** *: Cette erreur est lancée lorsque le programme* ***ReglesTypage*** *détecte que les bornes attribuées à la variable de contrôle ne sont pas de type integer. Elle indique* ***les types des deux variables de contrôle, le type demandé et affiche également le numéro de ligne associé à l'erreur***.

II – Architecture

Pour réaliser la passe 2, on parcourt récursivement l'arbre abstrait, on commence par décorer les déclarations, puis on décore les instructions.En effet la partie déclaration et instructions ne font pas appel aux mêmes fonctions puisque dans la première partie on vérifie l'existence des types et on met à jour la tables des symboles, alors que dans la partie instructions, on vérifie que les types indiqués sont respectés, que les arguments des opérations sont corrects et enfin que les identifiants utilisés font partie de la table des symboles.En cas d'erreur dans une affectation ou dans une opération, des erreurs de contextes sont lancées.

Verif.java s'occupe de l'interface entre la table des symboles et les vérifications de compatibilité faites par ***RèglesTypage.java***.

III-Méthodologie de test

Afin de réaliser les test les plus complets nous avons rédigé une liste la plus exhaustive possible de l'ensemble des cas d'affectations possibles (Valides et non valides) pour ensuite créer des fichiers de tests associés à la main car la génération par script de l'ensemble des tests nous aurait pris autant de temps.Il a été rédigé plus de 400 fichiers Jcas, de taille très variable, qui réalisent chacun un ou plusieurs tests.Il est important de notifier que certains tests ne font pas que tester la passe 2 mais font aussi des erreurs de syntaxe, pour vérifier par la même occasion la passe 1.

NOTA BENE : Avec nos tests, il arrive que certaines erreurs de passe 1 surviennent (erreurs Java) ,car certains tests présentent des erreurs de syntaxe, et certaines d'entre elles ne sont pas prises en compte en tant qu'erreur syntaxique mais en tant qu'erreur Java.

Voici la liste effectuée :

```
Vérification contextuelle, tests :
Erreurs possibles :

  Affectation :
    real := | real | +,-,* | int -> OK
                           | real -> OK
                           | Boolean -> ERR
                           | INTERVALLE -> OK
                           | Array -> ERR
                    | / | int -> OK
                        | real -> OK
                        | Boolean -> ERR
                        | INTERVALLE -> OK
                        | Array -> ERR
                    | and,or,not,div,mod,=,<,>,/=,<=,>=,[] -> ERR
                    | -> OK
            | int,INTERVALLE  | +,-,* | int -> OK
                                       | real -> OK
                                       | Boolean -> ERR
                                       | INTERVALLE -> OK
                                       | Array -> ERR
                              | / | int -> OK
                                  | real -> OK
                                  | Boolean -> ERR
                                  | INTERVALLE -> OK
                                  | Array -> ERR
                              | and,or,not,div,mod,=,<,>,/=,<=,>=,[] -> ERR
                              | -> OK
            | Boolean -> ERR
            | Array -> ERR
            | and,or,not,=,<,>,/=,<=,>=,[] -> ERR
            | +,- | int -> OK
                  | real -> OK
                  | Boolean -> ERR
                  | INTERVALLE -> OK
                  | Array -> ERR


    int,INTERVALLE :=  | real -> ERR
                       | int,INTERVALLE | +,-,* | int -> OK
                                                | real -> ERR
                                                | Boolean -> ERR
                                                | INTERVALLE -> OK
                                                | Array -> ERR
                                         | / -> ERR
                                         | and,or,not,=,<,>,/=,<=,>=,[] -> ERR
```

```
                                        | div,mod | int -> OK
                                                  | real -> ERR
                                                  | Boolean -> ERR
                                                  | INTERVALLE -> OK
                                                  | Array -> ERR
                                        | -> OK
                          | Boolean -> ERR
                          | Array -> ERR
                          |+,- | int,INTERVALLE -> OK
                               | real -> ERR
                               | Boolean -> ERR
                               | Array -> ERR
                          | and,or,not,div,mod,=,<,>,/=,<=,>=,[] -> ERR

Boolean := | real | =,<,>,/=,<=,>= | real -> OK
                                     | int -> OK
                                     | Boolean -> ERR
                                     | INTERVALLE -> OK
                                     | Array -> ERR
                        | +,-,*,/,and,or,not,div,mod,[] -> ERR
           | int | =,<,>,/=,<=,>= | real -> OK
                                  | int -> OK
                                  | Boolean -> ERR
                                  | INTERVALLE -> OK
                                  | Array -> ERR
                      | +,-,*,/,and,or,not,div,mod -> ERR
           | Boolean | and,or | real -> ERR
                               | int -> ERR
                               | Boolean -> OK
                               | INTERVALLE -> ERR
                               | Array -> ERR
                         | =,<,>,/=,<=,>=,+,-,*,/,not,div,mod -> ERR
                         | -> OK
           | INTERVALLE | =,<,>,/=,<=,>= | real -> OK
                                          | int -> OK
                                          | Boolean -> ERR
                                          | INTERVALLE -> OK
                                          | Array -> ERR
                           | +,-,*,/,and,or,not,div,mod,[] -> ERR
         | Array -> ERR
         | not | real | =,<,>,/=,<=,>= | real -> OK
                                        | int -> OK
                                        | Boolean -> ERR
                                        | INTERVALLE -> OK
                                        | Array -> ERR
                               | +,-,*,/,and,or,not,div,mod,[] -> ERR
              | int | =,<,>,/=,<=,>= | real -> OK
                                     | int -> OK
                                     | Boolean -> ERR
                                     | INTERVALLE -> OK
                                     | Array -> OK
                               | +,-,*,/,and,or,not,div,mod -> ERR
              | Boolean | and,or | real -> ERR
                                  | int -> ERR
                                  | Boolean -> OK
                                  | INTERVALLE -> ERR
                                  | Array -> ERR
                             |
              | INTERVALLE | =,<,>,/=,<=,>= | real -> OK
                                             | int -> OK
                                             | Boolean -> ERR
                                             | INTERVALLE -> OK
                                             | Array -> ERR
                                | +,-,*,/,and,or,not,div,mod,[] -> ERR
              | Array -> ERR
           | and,or,div,mod,=,<,>,/=,<=,>=,[],+,-,*,/ -> ERR
```