UFR 919 Sciences de l'ingénieur
Master Informatique – Spécialité STL
Analyse de Programmes et Sémantique (APS)

TD3 et 4 — Analyse statique et extension du langage

Jacques Malenfant, Olena Rogovchenko

1 Analyse Statique

Q1. Vérifier que l'exécution du corps d'une méthode se termine par un `return`, peut importe la séquence d'instructions effectivement exécutées.

Q2. Détecter toutes les lectures de variables locales non initialisées dans une instruction.

```
program
let
  Int i1, i2 ;
in
begin
  i1 := i2 + 5;
  writeln(i1)
end
```

Dans le programme ci-dessus, par exemple, on veut détecter que `i2` n'a pas été initialisée avant d'accèder à sa valeur.

2 Extension du langage

Q1. Rajouter au langage BOPL la boucle **for**, avec la syntaxe suivante :

`for` *id* `:=` *expr* `to` *expr* `step` *expr* `do` *inst*

Un exemple d'utilisation :

```
program
begin
  for i1 := 0 to 4 step 1 do
  begin
    writeln(i1)
  end
end
```

Q2. Rajouter à la boucle `while` l'instruction `break` qui permet de sortir de la séquence d'instructions, sans les exécuter.

Un exemple d'utilisation :

```
program
let
  Int i1;
in
begin
  i1 := 5 ;
  while true do
    begin
      if i1 = 0 then
        begin
          break
        end
      else
        begin
          i1 := i1 - 1
        end ;
      writeln(i1)
    end
end
```

3 Partie TME

Q1. Implanter en Prolog l'analyse vérifiant la présence de l'instruction `return` ; intégrer cette vérification dans le vérificateur de types.

Q2. Implanter en Prolog la détection des variables locales non-initialisées ; intégrer cette vérification dans le processus de vérification de types et d'évaluation.

Q3. Intégrer l'instruction `for` au vérificateur de types et à l'évaluateur. (*Nota : les analyseurs lexical et syntaxique qui vous sont fournis savent traiter la syntaxe proposée pour le* `for`.)

Q4. Intégrer l'instruction `break` au vérificateur de types et à l'évaluateur. (*Nota : les analyseurs lexical et syntaxique qui vous sont fournis savent traiter la syntaxe proposée pour le* `break`.)