

SE202 : projet de compilation Tiger

Samuel Tardieu
Année scolaire 2017/2018

Présentation du projet



Grands principes

- Tiger : langage de programmation “jouet” inventé par Andrew W. Appel pour ses livres “Modern compiler implementation in {C, Java, ML}”.
- Nous écrivons un compilateur d’un sous-ensemble de Tiger en C++.
- Certaines parties seront fournies (structures de données, squelettes), certaines ne pourront pas être modifiées.

Tiger



Tiger : le langage

- langage statiquement typé (nous nous contenterons pour l'instant du type `int` représentant des entiers sur 32 bits) ;
- permet de définir des variables, des fonctions (ce que nous implémenterons) ;
- permet de définir des types, des tableaux, des structures (ce que nous n'implémenterons pas).

Tiger : exemple de programme

Un programme Tiger sera composé pour nous d'une expression.

```
1 + 17
```

sachant qu'on peut définir des variables temporairement (de leur définition jusqu'à la fin du `end` correspondant)

```
let
  var a: int := 10  /* Type explicite */
  var b := 5 + a    /* Type implicite */
in
  a + b * 3
end
```

Tiger : exemple de let imbriqués

```
let
  var a := 5
in
  let
    var a := 6    /* Cache la définition précédente */
  in
    a + a        /* Vaut 12, le a le plus "proche" */
  end           /* est utilisé */
end
```

Vous aurez deviné que les commentaires sont compris entre /* et */. Ils peuvent être imbriqués.

Tiger : exemple de fonctions

Des fonctions peuvent être définies et peuvent être récursives :

```
let
  function fact(n: int): int =
    if n < 3
    then n
    else n * fact(n-1)
in
  fact(5)      /* 120 */
end
```


Tiger : le langage

- Les opérateurs binaires sont $+$, $-$, $*$, $/$, $\&$ (et logique), $|$ (ou logique) et sont associatifs à gauche.
- Les opérateurs binaires de comparaison sont $<$, $<=$, $>$, $>=$, $=$, $<>$ (différent) et ne sont pas associatifs
- Du plus prioritaire au moins prioritaire, les ordres sont (par groupe) : les parenthèses, le $-$ unaire, $*$ et $/$, $+$ et $-$ (binaire), les opérateurs de comparaison, $\&$, puis $|$.
- Il existe un opérateur unaire $-$.
- Le résultat d'une comparaison entre deux valeurs renvoie soit 1 (comparaison vraie) soit 0 (comparaison fausse).
- Lors d'un test dans `if`, toute valeur différente de 0 est vraie, 0 étant la valeur fausse.

Tiger : le langage

- Les mots clés sont `array`, `break`, `do`, `else`, `end`, `for`, `function`, `if`, `in`, `let`, `nil`, `of`, `then`, `to`, `type`, `var` et `while`.
- Les identifiants commencent par une lettre (majuscule ou minuscule) qui peut être suivie de lettres, chiffres ou *underscores* (`_`). Ils ne doivent pas correspondre à un mot clé. La châsse (*case*) est significative, `a` et `A` sont deux identifiants différents.
- Les nombres sont des entiers en décimal qui ne doivent pas comporter de 0 initiaux inutiles.
- La déclaration des types n'est obligatoire que dans les paramètres des déclarations de fonction, et facultatifs pour la déclaration de variable et le type de retour des déclarations de fonctions.

Tiger : le langage

Exemples :

```
let
  var abc12_34x := 0      /* Autorisé */
  var abc12_34x: int := 0 /* Autorisé */
  var _abc := 0          /* Interdit */
  var abc := 012         /* Interdit */
  var in := 0           /* Interdit */
in
  ...
end
```