Module Sémantique
TD 3 : Continuations et réponses

1 Notion de réponse

1. Qu'est ce que la réponse d'un programme ?

2. On prendra ici pour réponse la suite des valeurs imprimées à l'écran.

2. 1. Donner la sémantique par continuation de l'instruction `Print(e)` où `e` est une expression.

2. 2. Quelle est la continuation que l'on utilisera pour calculer la sémantique d'un programme P ?

2. 3. Calculer la sémantique par continuations du programme

```
x := 2;
y := 1;
Print(z := x + 3 * y);
Print(x++);
```

3. On considère maintenant pour réponse la valeur de la variable `x`.

3. 1. Quelle est la continuation que l'on utilisera pour calculer la sémantique d'un programme P ?

3. 2. Calculer la sémantique par continuations du programme

```
x := 2;
y := 1;
Print(z := x + 3 * y);
Print(x++);
```

2 Sauvegarde d'environnement

La notion de réponse considérée dans cette partie sera la suite des valeurs imprimées à l'écran.

4. Donner la sémantique par continuation des instructions `try P with a -> Q` et `raise a`.

5. Quelle sera la réponse de l'évaluation de la sémantique par continuation du programme suivant ?

```
try
   x := 1;
   Print(x);
   try
      x := 2;
      Print(x);
      raise a;
      x := 3;
      Print(x);
   with
      a -> raise a;
   x := 4;
   Print(x);
with
   a -> x := 5;
        Print(x);
```

On veut maintenant permettre aux utilisateurs de Toy la syntaxe `{...}` pour délimiter des parties *closes* du code dans lesquelles on pourra utiliser des variables locales.

6. Proposer une sémantique par continuations pour la syntaxe de blocs `{...}`.

7. Calculer la sémantique par continuations du programme

```
x := 2;
y := 1;
  { x := x + 6;
    y := x + 2;
    Print(y); }
Print(x++);
```

8. Que dire de la sémantique par continuation de `{...}` dans le cas où la réponse désirée est la valeur d'une variable (la variable `x` par exemple) ? Que dire alors de l'évaluation de la sémantique du programme ci-dessus ?