

Calculabilité & Complexité [M1]

Hugo Férée, Valia Mitsou

Feuille de TD n°3
Théorèmes de récursion

Exercice 1 (Théorème de Rice) *Démontrez-le.*

Exercice 2 *Appliquer le théorème de Rice pour montrer que les langages suivants sont indécidables :*

- $INFINITE_{TM}$
- $EXISTS_{TM}$
- REG_{TM}
- $HALT_0$

Trouver trois exemples d'analyses statiques de programmes impossibles à implémenter à cause du théorème de Rice.

Exercice 3 (Théorème du point fixe) *Montrer comment utiliser le théorème du point fixe pour définir une fonction récursive, comme dans un langage fonctionnel. On pourra utiliser la fonction fibonacci comme exemple.*

Exercice 4 (Théorème de récursion) *Utiliser le théorème de récursion pour (re)prouver les résultats suivants :*

1. *Le problème de l'arrêt est indécidable.*
2. *Le théorème de Rice.*
3. *On dit qu'une machine est minimale si parmi les machines équivalentes, elle est celle la plus petite (en terme de taille de code). Montrer que le langage des (codes de) machines minimales n'est pas semi-décidable.*

Exercice 5 (Quines) 1. *Écrivez un quine en français.*

2. *Écrivez un quine en Ocaml (utiliser `Printf.printf` sachant que le format `"%S"` affiche une chaîne de caractères entre guillemets).*

3. *Complétez le quine suivant, écrit en python3 :*

```
a = [ '...', 'for s in a:\n    print(s)']  
...
```