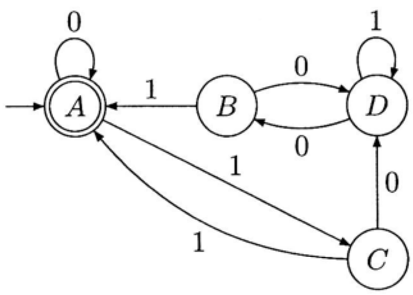
ENSEIRB - Informatique - 1ère année 2003-2004

# Examen du cours - 2ème session
# «Analyse d'algorithmes»
# (Sans documents)

**Exercice 1 (Langages réguliers)**

1. Indiquez, en justifiant, une méthode permettant de décider si deux automates finis acceptent le même langage.
2. En vous basant sur cette méthode, indiquez si l'automate suivant accepte le langage des entiers naturels multiples de 3, encodés en binaire. Vous reproduirez l'intégralité du calcul.




**Exercice 2 (QCM)**

Indiquez, en justifiant vos réponses, si selon vous les affirmations suivantes sont vraies ou fausses. **Attention, cet exercice est un QCM : 1pt par réponse correcte, 0pt en absence de réponse, -1pt par réponse incorrecte ou incorrectement justifiée.**

1. Il existe au moins un langage qui est accepté par une machine de Turing non-déterministe, mais par aucune machine de Turing déterministe.
2. La méthode de Hoare permet de prouver la correction partielle de n'importe quel while-program.
3. Il est possible de prouver la terminaison de n'importe quel while-program par la méthode des ensembles bien fondés.
4. Un langage accepté par une machine de Turing déterministe à $k > 1$ rubans de complexité $f(n) = n + 1$ est accepté par une machine de Turing déterministe à 1 ruban de complexité asymptotique $O(n^2)$.

**Exercice 3 (Preuve de correction)**

Soit le while-program suivant, où `a%d` est le reste de la division de `a` par `d` :

```
si (a=0) ou (a=1) alors
  d := a
sinon
  d := a-1 ;
  tantque (d >= 2) et (a%d != 0) faire
    d := d-1
  fintantque
finsi
```

Pour tout entier naturel a, cet algorithme calcule d, le plus grand entier naturel diviseur de a strictement inférieur à a. Par convention, d = 0 lorsque a = 0 et d = 1 lorsque a = 1. Nous avons donc :

- PRECOND : $\{a \in \mathbb{N}\}$
- POSTCOND : $\{(\exists i \in \mathbb{N}.\, a = d \times i) \wedge (\forall d' \in \mathbb{N}.d < d' < a,\ a\%d' \neq 0)\}$

Prouver la correction de ce while-program en utilisant la démarche suivante :

1. On ne considère pour le moment que les lignes 4 à 7.
   (a) que peut-on dire de la valeur de a en ligne 4 ? Que peut-on prouver sur celle de d ?
   (b) à partir de ces propriétés sur les valeurs de a et d, déterminer :
      i. un invariant de la boucle `tantque` qui permet de conclure la POSTCOND donnée précédemment.
      ii. un terme que vous utiliserez comme mesure de décroissance pour la terminaison de la boucle `tantque`.
   (c) prouver la correction totale de la boucle `tantque`.
   (d) prouver la correction totale des lignes 4 à 7 : en partant des PRECOND et des propriétés sur la valeur de a obtenues en 1a, déduisez les POSTCOND.
2. On considère maintenant la ligne 2.
   (a) que peut-on dire de la valeur de a ?
   (b) à partir des PRECOND et des propriétés sur la valeur de a, prouver la correction totale (c'est à dire les POSTCOND) de la ligne 2.
3. Prouvez finalement la correction totale du while-program.

## Règles du calcul de Hoare (preuve de correction totale) :

Règle de conséquence :

$$\frac{P' \Rightarrow P,\ [P]\,S\,[Q]\ ,\ Q \Rightarrow Q'}{[P']\,S\,[Q']}$$

Règle d'affectation :

$$\frac{}{[P[e/x]]\,x{:=}e\,[P]}$$

Règle de séquence :

$$\frac{[P]\,S_1\,[Q]\ ,\ [Q]\,S_2\,[R]}{[P]\,S_1;S_2\,[R]}$$

Règle du «`si`» :

$$\frac{[P \wedge B]\,S_1\,[R]\ ,\ [P \wedge \neg B]\,S_2\,[R]}{[P]\,\texttt{si}\ B\ \texttt{alors}\ S_1\ \texttt{sinon}\ S_2\ \texttt{finsi}\,[R]}$$

Règle du «`tantque`» :

$$\frac{[P \wedge B \wedge (t = X)]\,S\,[P \wedge (t < X)]\ \wedge (P \wedge B \Rightarrow t \in \mathbb{D})}{[P]\,\texttt{tantque (B) faire S fintantque}\,[P \wedge \neg B]}$$

où :
- $X$ est une variable qui n'apparaît pas dans $P$, $B$, $t$ et $S$.
- $t$ est un terme dont les variables libres sont les variables de $P$, $B$ et $S$.
- Il existe une relation d'ordre $\leq$ sur $\mathbb{D}$ telle que $(\mathbb{D}, \leq)$ est un ensemble bien fondé.
- $t = X$ est équivalent à $(t \leq X) \wedge (X \leq t)$.
- $t < X$ est équivalent à $\neg(X \leq t)$.