

FONDEMENTS DE LA PROGRAMMATION

MASTER 1 INFORMATIQUE 2017-2018
INSTITUT GALILÉE - UNIVERSITÉ PARIS 13

Paulin de Naurois - Domenico Ruoppolo
(d'après un cours par Virgil Mogbil et Pierre Boudes)

TD 3: MACHINE À ADRESSAGE INDIRECT CONCURRENTTE (CRAM)

Remarque: à partir de maintenant vous êtes autorisé.e.s à utiliser non seulement les macros vues en séance précédente pour les SRAM, c'est-à-dire `goto ℓ` | `Xi := constante` | `Xi := Xj`, mais aussi les instructions suivantes, également dérivables dans toute SRAM:

`Xi := Xj + Xz` | `Xi := Xj - Xz` | `if Xi = Xj goto ℓ else ℓ'` | `if Xi < Xj goto ℓ else ℓ'` | `if Xi ≤ Xj goto ℓ else ℓ'` | `if Xi = constante goto ℓ else ℓ'` | `if Xi < constante etc.`

Exercice 1. Décrire une CRAM qui teste si a appartient à une suite d'éléments $(a_i)_{1 \leq i \leq n}$.

Exercice 2. Décrire une CRAM qui prend en entrée un nombre naturel n et une suite d'autres m naturels $(a_i)_{1 \leq i \leq m}$ et qui rend la suite $(n \cdot a_i)_{1 \leq i \leq m}$. On placera l'entrée n dans le compteur d'index 0 de la mémoire globale et les autres entrées a_1, \dots, a_m dans les registres suivants de la même mémoire. Les résultats seront placés dans les registres de la mémoire globale à partir de l'index 1. Donc formellement la CRAM donnera en sortie: $n, na_1, na_2, \dots, na_m$.

Exercice 3. Décrire une CRAM qui prend en entrée un nombre naturel n , deux suites de naturels $(a_i)_{1 \leq i \leq n}$ et $(b_i)_{1 \leq i \leq n}$ de longueur n , et qui rend $\sum_{i=1}^n i a_i + (n+1-i) b_i$. Par exemple, pour $n = 3$ l'on retrouvera $a_1 + 3b_1 + 2a_2 + 2b_2 + 3a_1 + b_3$ en sortie (i.e. dans le compteur G_0).

Exercice 4. Décrire une CRAM qui vérifie l'égalité de deux mots de naturels $(a_i)_{1 \leq i \leq n}$ et $(b_i)_{1 \leq i \leq m}$. On suppose qu'en entrée leur tailles n et m soient dans les registres G_0 et G_1 de la mémoire globale, suivies par a_1, \dots, a_n et ensuite b_1, \dots, b_m .

Exercice 5. Décrire une CRAM qui calcule le maximum des valeurs $(a_i)_{1 \leq i \leq n}$.

Aide: pour toute couple (i, j) , pour $1 \leq i, j \leq n$, l'on considère un processeur qui compare a_i avec a_j et écrit le résultat de cette comparaison dans un "tableau de booléens" (initialisé à la valeur vraie partout). Utilisez donc n^2 processeurs concurrents!

Exercices complémentaires. Faire les exercices de cette fiche avec des SRAM. Et *viceversa* faire les exercices du précédent TD en utilisant des CRAM.