

## Circuits et Architecture (CA7)

### TD n° 9 : Utilisation d'un simulateur LC-3

Durant cette séance de travaux pratiques, nous allons utiliser `lc3sim`, un programme dédié à la simulation du processeur LC-3. Ce simulateur va permettre de mettre au point des programmes LC-3 plus complexes, qui pourront par la suite servir de tests lorsque votre circuit `logisim` implémentant le LC-3 sera plus complet.

Le simulateur `lc3sim` se trouve dans le dossier `tools/lc3` du dépôt git du cours<sup>1</sup>. Le code est accompagné d'un fichier `README` expliquant comment le compiler. Il provient des outils disponibles sur le site du livre de Patt et Patel<sup>2</sup>.

Une fois les `lc3tools` compilés, le simulateur `lc3sim` peut être lancé depuis le terminal. Ce simulateur accepte des fichiers `.obj`, soit directement en argument sur la ligne de commande, soit après lancement dans la case "*File to load*". Pour transformer un fichier source assembleur `.asm` en version binaire `.obj`, utiliser le programme d'assemblage `lc3as`. Au passage, `lc3as` a été modifié pour produire également un fichier mémoire `.mem` utilisable sous `logisim`.

*Nota bene* : dans vos programmes assembleur, faites attention à écrire `x3000` pour une adresse en hexadécimal derrière la directive `.ORIG`. Pour nos petits programmes, on peut aussi se contenter de les placer à l'adresse 0.

#### Exercice 1 – Simulation avec `lc3sim`

Récupérez le programme de la multiplication naïve non signée `MultNaive.asm` sur la page du cours. Assemblez-le en un fichier `MultNaive.obj` grâce à `lc3as`. Testez-le ensuite sur `lc3sim`.

#### Exercice 2 – Somme partielle

Écrivez un programme assembleur LC-3 qui calcule les sommes partielles d'une suite de 10 entiers écrits en mémoire (à la fin du programme), en écrasant cette même suite en mémoire. Par exemple, (5,2,10,3,6,120,120,10,0,1) est remplacée par (5,7,17,20,26,146,266,276,276,277). Simulez-le sous `lc3sim`.

#### Exercice 3 – Sous-routines

On veut faire un programme qui renverse le contenu d'une chaîne de caractères en place. Pour cela, on utilisera une sous-routine de calcul de la longueur d'une chaîne (vue en cours) et une sous-routine qui échange le contenu de 2 adresses mémoire.

#### Exercice 4 – Pile

Écrivez un programme qui calcule la factorielle d'un nombre de manière récursive. Le nombre sera lu en mémoire et le résultat écrit dans la même mémoire à la fin. On pourra utiliser la multiplication naïve comme sous-routine.

1. <https://gaufre.informatique.univ-paris-diderot.fr/aguatto/architecture-m1-22-23>

2. Une version pour Microsoft Windows est disponible sur le site du livre, suivez le premier lien "LC-3 Simulator", par contre il faudra faire sans nos petites améliorations.