

Travaux Dirigés de Compilation n°5

Licence d'informatique

Fonctions en assembleur

Le but de ce TD est d'apprendre à implémenter des fonctions en assembleur et à se familiariser avec les conventions AMD 64.

► Exercice 1. *Les conventions d'appel AMD64*

Écrivez un ou plusieurs exemples de code *C* tout petits. On obtient du code en assembleur en lançant la commande `g++ -S -masm=intel monFichier.c` ou la même chose avec `gcc`. Observez comment l'assembleur passe les arguments, accède aux arguments et renvoie la valeur de retour.

Remarques

- Dans le résultat, les accès à la mémoire peuvent contenir le mot-clé `PTR`. Ce mot-clé n'est pas conforme à la documentation de `nasm` et on peut le supprimer, mais certains outils le tolèrent.
- L'instruction `leave` libère le bloc d'activation.

► Exercice 2. *Arguments et valeur de retour*

1. Écrivez en assembleur une fonction qui prend un entier signé codé sur 4 octets et qui renvoie son carré. Respectez les conventions d'appel AMD 64.

Complément de cours sur l'instruction `imul` : si on utilise l'instruction `imul` avec des opérands sur 4 octets, elle tronque le résultat à 4 octets.

2. Écrivez dans un autre fichier du code en assembleur qui appelle votre fonction.

► Exercice 3. *Registres volatils et non volatils*

Écrivez une fonction qui prend l'adresse de deux entiers, codés sur 4 octets chacun, et qui les intervertit. Respectez les conventions d'appel AMD 64, y compris en ce qui concerne les registres volatils et non volatils. Testez en appelant votre fonction depuis du code en `nasm`.

► Exercice 4. *Alignement de la pile*

Écrivez en assembleur une fonction récursive qui prend en arguments deux entiers positifs ou nuls m et n et qui renvoie mn en utilisant `add` mais ni `imul` ni `mul`. Respectez les conventions d'appel AMD 64, y compris en ce qui concerne l'alignement de la pile. Testez en appelant la fonction depuis du code en `nasm`.