

Le compilateur est plus fort que toi

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 13 avril 2010

<https://www.bortzmeyer.org/source-code-optim.html>

L'exposé « *Source Code Optimization* » <http://www.linux-kongress.org/2009/slides/compiler_survey_felix_von_leitner.pdf> » de Felix von Leitner (contributeur entre autres à mutt) expose la vanité qu'il y a à optimiser le code C à la main alors que le compilateur fait mieux dans la plupart des cas.

L'exemple de la page 37 - le programmeur a fait un décalage pour optimiser une division - est particulièrement simple, mais celui de la page 42, tiré d'un codec vidéo réel, est encore plus éclairant.

L'auteur teste du code C avec plusieurs compilateurs (comme gcc, llvm, Intel C ou Sun C) et compare le code assembleur produit. Il est presque toujours meilleur que ce que le programmeur normal produirait lorsqu'il essaie d'« optimiser ».

À noter qu'il ne semble pas y avoir un compilateur radicalement meilleur, chacun est plus fort sur certaines optimisations et pas sur d'autres.

Attention, c'est long et compliqué (ce sont des transparents, le code assembleur n'est pas toujours expliqué). Il faut bien réviser son assembleur x86 avant.

Tiens, un petit test : quelles sont les quatre façons possibles, en assembleur, de mettre zéro dans un registre (p. 70)? Je donne la plus facile, `mov $0, %eax`, à vous de trouver les trois autres. Bonus : prédire quelle est la plus efficace.

Merci à Mathieu Arnold pour le titre de cet article.