

Integração C-Assembly “às escondidas”: C/C++ Intrinsic

Integração C-Assembly

- Contexto
 - Anteriormente vimos 2 formas possíveis de integração C-Assembly
 - Código assembly “inline”
 - Direção “C usando assembly”
 - Utilização de arquivos assembly separados, seguindo o padrão das “convenções C” (C Calling Conventions)
 - Direção “C usando assembly” e
 - Direção “assembly usando C”

Integração C-Assembly

- Vantagens das formas anteriormente citadas
 - Abre-se todo o leque de opções em assembly para o programador
- Desvantagens
 - Abre-se todo o leque de opções em assembly para o programador

Integração C-Assembly

- Vantagens das formas anteriormente citadas
 - Abre-se todo o leque de opções em assembly para o programador
- Desvantagens
 - Abre-se todo o leque de opções em assembly para o programador, digo, o programador tem que dominar assembly

Intrinsics

- Ideia geral
 - Fornecer uma API em C para instruções Assembly (instruções SIMD, sobretudo)
 - Veja que isso, de certa forma, é o que uma linguagem de alto nível (tipo C) faz para o conjunto de instruções “comum”
 - Muitas vezes trata-se de um mapeamento 1-1 entre intrinsic e instrução assembly
 - (mas não necessariamente)

Exemplo: Implementações

- Intel C++ Intrinsic
 - <http://goo.gl/915A9I>
- Introdução a Intrinsic no GCC
 - <http://goo.gl/9BGJpL>

Atividade: usando intrinsics

- Estude e entenda o código
intrinsic/samplecode/simple/x86/temperatures
_sse.c e
intrinsic/samplecode/simple/x86/temperatures
_mmx.c dentro de intrinsic.zip
 - (parte do código exemplo do segundo link do slide anterior, também disponível em <http://goo.gl/f1CN5O>)