

山东大学 计算机科学与技术 学院

汇编语言 课程实验报告

学号：202200130048	姓名：陈静雯	班级：6 班
实验题目：GCC 内联汇编优化		
实验学时：2	实验日期：20241202	
实验目的：掌握 AT&T 语法下的 AMD64 汇编编写。掌握 SIMD 指令的使用，并对程序进行向量化优化。掌握 C 与内联汇编联合编程的方法。		
实验环境：Windows10、mingw-64		
源程序清单： 1. Vector.c		
编译及运行结果：  <pre>PS D:\czsystem\huibian> gcc vector.c -o vector.exe</pre> 运行结果：  <pre>PS D:\czsystem\huibian> ./vector.exe naive - 2751463424 - 43674781995 cycles expert - 2751463424 - 31060319923 cycles sse - 2751463424 - 9790971199 cycles avx-auto - 2751463424 - 4299216174 cycles avx-manual - 2751463424 - 2993860617 cycles PS D:\czsystem\huibian></pre>		

问题及收获：

1. 使用 mingw-w64 作为编译器。该编译器内联汇编的语法与非内联汇编有何不同，又与 MASM 有何不同？

(1) 内联汇编：在 C/C++ 代码中直接嵌入汇编指令，这些指令通常被放置在函数体内，并且可以访问周围的 C/C++ 变量。MinGW-w64 使用 GCC 风格的内联汇编语法，它允许指定输入操作数、输出操作数和修改的寄存器等。

(2) 非内联汇编：指的是用汇编语言单独编写的程序或函数，然后链接到 C/C++ 程序中。这种方式需要编写完整的汇编源文件（.s 或 .asm），并且遵循特定的汇编语法，如 AT&T 或 Intel 语法。非内联汇编提供了更精细的控制，但需要更多的工作来维护和集成。

(3) MinGW-w64 内联汇编 vs MASM：两者在语法风格、操作数顺序、寄存器命名、立即数前缀、内存引用等方面有显著不同。MinGW-w64 内联汇编更适合与 C/C++ 代码集成，而 MASM 更适合编写完整的汇编程序。

2. 内联汇编的 clobber list(破坏列表)有何用途？能否不写？

(1) 防止寄存器冲突：编译器需要知道哪些寄存器可能会被内联汇编代码修改，以避免将重要的值存储在这些寄存器中。如果不指定 clobber 列表，编译器可能会错误地认为某些寄存器是未修改的，从而导致程序行为异常。

(2) 确保正确的寄存器分配：编译器会根据 clobber 列表调整寄存器分配策略。例如，如果内联汇编代码修改了某个寄存器，编译器会确保在内联汇编之前保存该寄存器的内容，并在之后恢复它。

(3) 处理内存访问：如果内联汇编代码可能会修改内存（例如通过指针访问），你需要在 clobber 列表中添加 "memory"。这告诉编译器不要假设任何内存位置在内联汇编执行前后保持不变。这对于防止编译器进行不安全的优化非常重要。

(4) 可以不写 Clobber List，但有以下几点问题：

编译器优化问题：如果不指定 clobber 列表，编译器可能会做出错误的假设，导致生成的代码在某些情况下出现错误。例如，编译器可能会认为某些寄存器没有被修改，从而在内联汇编前后没有正确保存和恢复这些寄存器的值。

潜在的未定义行为：如果你没有正确声明被修改的寄存器或内存，编译器可能会生成不正确的代码，导致程序行为不可预测，甚至崩溃

3. 比较并评价 SSE, AVX2 (与 AVX-512, 有条件的同学可尝试) 向量化指令的性能。能得到什么结论? 为什么?

SSE (Streaming SIMD Extensions)、AVX2 (Advanced Vector Extensions 2) 和 AVX-512 是 Intel 提供的向量扩展指令集, 用于加速浮点和整数运算。这些指令集通过并行处理多个数据元素来提高计算效率。

(1) AVX-512 拥有最宽的向量宽度, 可以在单次指令中处理更多的数据元素, 理论上具有最高的吞吐量。AVX2 次之, SSE 最低。

(2) AVX-512 不仅拥有更宽的寄存器, 还提供了更多的寄存器, 这有助于减少寄存器溢出到内存的情况, 从而提高性能。更多的寄存器意味着编译器可以在寄存器中保存更多的中间结果, 减少内存访问的开销。

(3) AVX-512 提供了更多高级指令, 特别适合复杂的科学计算、机器学习和深度学习等应用场景。AVX2 也引入了一些有用的整数向量化指令, 但 AVX-512 在灵活性和功能上更为强大。