Exam Assignments V07

chengchengguo 183090

1. Explain **three vectorization clauses** of your choice that can be used with **#pragma omp simd**.

   1) align clause
   aligned(*list*[:*alignment*])
   Declares that the objects in *list* are byte aligned according to the number of bytes expressed by *alignment*.
   2) simdlen clause
   The **simdlen** clause is a **hint** to guide the compiler in the selection of a **vector length** for the loop. (Intel Compiler adheres to the specification of simdlen.)
   e.g. *simdlen(8) --> eight floats per vector register*
   (Usually it is not necessary to specify simdlen.)
   3) safelen clause
   The **safelen** clause **sets** a **vector width limit** (can be useful for dependencies between loop iterations).

   For example, **safelen(16)** means that the **loop can be vectorized safely** with a **vector length of 16 or less**.

   Note the difference between the safelen and simdlen clauses. The **safelen** clause is **required for correctness** while the **simdlen** clause **indicates a preference**.

2. Give **reasons** that speak **for and against vectorization with intrinsics** compared to **guided vectorization with OpenMP**.

   for:
   Intrisics provide **data types** for **vectors**, and, **functions** that operate directly on **vectors**.

   against:
   major disadvantage: **intrinsic data types** and **intrinsic functions** are **specific** for a **particular processor** architecture
   ➔ the gain of performance portability comes with a loss of code portability

3. What are the **advantages** of **vector intrinsics over assembly** code?

➔ intrinsic functions have the advantage that **explicit assignments of registers** are **omitted**
➔ intrinsic functions are **more portable** between different compilers and operating systems (valid **C++ code**)
➔ **vector intrinsics** are basically **wrappers around** corresponding **assembly** instructions
➔ **easier** to **learn**, **easier** to **use** and **more readable code** than assembly

Intrinsics provide the following benefits: [1]

- Powerful. Intrinsics allow the programmer to inline assembly code without having to explicitly program in assembly.
- Portable. Code containing intrinsics can be compiled for different SVE enabled platforms by setting the `-mcpu` option. However, this may not yield the same performance.
- Flexible. The programmer can use intrinsics or C/C++ code in the same program.

4. What are the corresponding **vectors** of the **three intrinsic data types**: **__m256**, **__m256d** and **__m256i**.

➔ __m256 (for eight floats)
➔ __m256d (for four doubles)
➔ __m256i (for integers, no matter the size)

[1] SVE Optimization Guide (arm.com)