Задание к лабораторной работе №6

* 1. Найти описание своего процессора в интернете, выяснить, какие SSE-расширения он поддерживает (SSE, SSE2, SSE3, SSE4, AVX, AVX512) Изучить архитектуру расширений .
  2. Реализовать алгоритм работы с матрицами в ассемблерной вставке Visual Studio C++ без использования SSE (AVX)3
  3. с использованием SSE (AVX)
  4. **сравнить время**  выполнения (для измерения времени использовать rdtsc)
  5. Внимание! после вариантов задания приводится описание некоторых нужных команд SSE

Варианты

1.

Размерность матриц 4\*4.

Увеличить все элементы матрицы М1 на заданное число. Умножить М1 на матрицу М2. Сложить поэлементно первую и n-ю строки матрицы

2.

Размерность матриц 8\*8.

Умножить все элементы матрицы М1 на заданное число. Умножить М1 на матрицу М2. Сложить поэлементно первый и n-й столбцы матрицы

3.

Размерность матриц 4\*4.

Уменьшить все элементы матрицы М1 на заданное число. Умножить М1 на матрицу М2. Сложить поэлементно М1 и матрицу-произведение.

4.

Размерность матриц 8\*8.

Сложить поэлементно матрицы М1 и М2. Умножить полученную матрицу на матрицу М2. Умножить все элементы полученной матрицы на заданное число.

5.

Размерность матриц 4\*4.

Сложить поэлементно первый и 2-й столбцы матрицы М1. Умножить полученную матрицу на матрицу М2. Умножить все элементы полученной матрицы на заданное число.

6.

Размерность матриц 8\*8.

Сложить поэлементно первую и 3-ю строки матрицы М1. Результат записать в первую строку. Умножить полученную матрицу на матрицу М2. Умножить все элементы полученной матрицы на заданное число.

7.

Размерность матриц 4\*4.

Уменьшить все элементы матрицы М1 в заданное число раз. Умножить М1 на матрицу М2. Сложить поэлементно М1 и матрицу-произведение

8.

Размерность матриц 8\*8.

Сложить поэлементно матрицы М1и М2. Результат записать в первую матрицу. Умножить полученную матрицу на матрицу М2. Умножить все элементы полученной матрицы на заданное число.

9.

Размерность матриц 4\*4.

Сложить поэлементно первый и 3-й столбцы матрицы М1. Умножить полученную матрицу на матрицу М2. Поделить все элементы полученной матрицы на заданное число.

10.

Размерность матриц 8\*8.

Уменьшить все элементы матрицы М1 в заданное число раз. Умножить М1 на матрицу М2. Сложить поэлементно М1 и матрицу-произведение. . Сложить поэлементно первый и n-й столбцы матрицы.

11.

Размерность матриц 4\*4.

Уменьшить все элементы матрицы М1 вдвое. Умножить М1 на матрицу М2. Сложить поэлементно первую и 2-ю строки матрицы

12.

Размерность матриц 8\*8.

Умножить все элементы матрицы М1 на 2. Умножить М1 на матрицу М2. Сложить поэлементно первый и n-й столбцы матрицы

**Команда SHUFPS xmm2, xmm1/m, imm**

Переставляет числа в регистре XMM1 в соответствии со значением imm. Двухбитные значения из imm означают номера 32-битных чисел из XMM1. Порядок их расположения в imm показывает порядок размещения этих чисел в XMM2. Пусть imm = 01 00 10 11. Тогда числа будут перемещаться следующим образом:

XMM1:

|  |  |  |  |
| --- | --- | --- | --- |
| 11 | 10 | 01 | 00 |

XMM2:

|  |  |  |  |
| --- | --- | --- | --- |
| 01 | 00 | 10 | 11 |

**Команда HADDPS xmm1,xmm2**

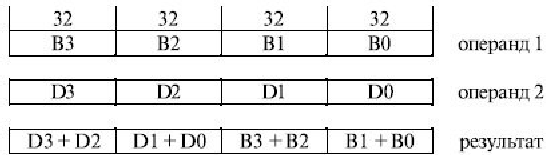
осуществляет горизонтальное сложение элементов с одинарной точностью.

Первый результат является суммой первого и второго элементов первого (исходного) операнда;

второй результат - суммой третьего и четвертого элементов первого операнда;

третий результат - суммой первого и второго элементов второго операнда (операнда назначения)

и, наконец, четвертый результат - суммой третьего и четвертого элементов второго операнда.



**Команда MOVAPS xmm/m, xmm/m –**

команда копирует четыре FP-элемента одним из следующих способов:

* из памяти в XMM-регистр
* из XMM-регистра в память
* из одного XMM-регистра в другой

Линейный адрес соответствует адресу младшего байта группы данных в памяти. Обращение в память должно быть по адресу, кратному 16 байтам; в противном случае генерируется исключение.

Выравнивание матриц в памяти в с++:

\_declspec(align(16)) float m1[n][n];

**Команда MOVUPS xmm/m, xmm/m –**

команда копирует четыре FP-элемента одним из следующих способов:

* из памяти в XMM-регистр
* из XMM-регистра в память
* из одного XMM-регистра в другой

Линейный адрес соответствует адресу младшего байта группы данных в памяти. Команда применяется для обращения к невыровненным данным в памяти.