COMP2006 Computer Organization

AVX Intrinsics Exercise Worksheet

# Exercise

1. Multiply two sets of values and print the result to the console.

/// put your code here

//Q1

#include <immintrin.h>

#include <stdio.h>

int main()

{

    int i;

    double A[] = {3.3, 1.5, 4.4, 2.2, 9.5, 1.4, 5.4, 4.3, 3.8, 5.8, 8, 9.3, 8.5, 0.8, 5.3, 4.3, 5.6, 4.5, 1.2, 2.1};

    double B[] = {9.6, 4.7, 8.7, 8.4, 5.3, 1.3, 7.8, 6.1, 7.5, 0.1, 5.4, 3.4, 6.3, 7.4, 2.9, 0.2, 1.7, 5.9, 1.2, 7.4};

    double C[20] = {0.0};

    \_\_m256d vA;

    \_\_m256d vB;

    for( i =0; i<20; i=i+4){

  vA =\_mm256\_load\_pd(A + i);

     vB =\_mm256\_load\_pd(B + i);

      \_\_m256d vC =\_mm256\_mul\_pd(vA, vB);

    \_mm256\_store\_pd(C+i, vC);

    }

   for(int i=0;i<20;i++){

    printf("C[%d] = %.1lf\t", i, C[i]);

    }

    return 0;

}

1. Divide two sets of values and print the result to the console.

/// put your code here

//Q2

#include <immintrin.h>

#include <stdio.h>

int main()

{

    int i;

    double A[] = {3.3, 1.5, 4.4, 2.2, 9.5, 1.4, 5.4, 4.3, 3.8, 5.8, 8, 9.3, 8.5, 0.8, 5.3, 4.3, 5.6, 4.5, 1.2, 2.1};

    double B[] = {9.6, 4.7, 8.7, 8.4, 5.3, 1.3, 7.8, 6.1, 7.5, 0.1, 5.4, 3.4, 6.3, 7.4, 2.9, 0.2, 1.7, 5.9, 1.2, 7.4};

    double C[20] = {0.0};

    \_\_m256d vA;

    \_\_m256d vB;

    for( i =0; i<20; i=i+4){

  vA =\_mm256\_load\_pd(A + i);

     vB =\_mm256\_load\_pd(B + i);

      \_\_m256d vC =\_mm256\_div\_pd(vA, vB);

    \_mm256\_store\_pd(C+i, vC);

    }

   for(int i=0;i<20;i++){

    printf("C[%d] = %.1lf\t", i, C[i]);

    }

    return 0;

}

1. Divide the values of the set A by two and print the result to the console.

/// put your code here

//Q3

#include <immintrin.h>

#include <stdio.h>

int main()

{

    int i;

    double A[] = {3.3, 1.5, 4.4, 2.2, 9.5, 1.4, 5.4, 4.3, 3.8, 5.8, 8, 9.3, 8.5, 0.8, 5.3, 4.3, 5.6, 4.5, 1.2, 2.1};

    double B[] = {9.6, 4.7, 8.7, 8.4, 5.3, 1.3, 7.8, 6.1, 7.5, 0.1, 5.4, 3.4, 6.3, 7.4, 2.9, 0.2, 1.7, 5.9, 1.2, 7.4};

    double two[] = {2};

    double C[20] = {0.0};

    \_\_m256d vA;

    \_\_m256d vB;

    for( i =0; i<20; i=i+4){

  vA =\_mm256\_load\_pd(A + i);

     vB =\_mm256\_broadcast\_sd(two);

      \_\_m256d vC =\_mm256\_div\_pd(vA, vB);

    \_mm256\_store\_pd(C+i, vC);

    }

   for(int i=0;i<20;i++){

    printf("C[%d] = %.1lf\t", i, C[i]);

    }

    return 0;

}