

**Course Name**: GPU Parallelism and Performance

**Course Number and Section**: **14:332:333:xx**

**Experiment**: [Experiment # 6 – GPU Parallelism and Performance]

**Lab Instructor**: Tuan-Anh Le

**Date Performed**: 11/29/16

**Date Submitted**: 12/9/16

**Submitted by**: Peter Luo 157004870

--------------------------For Lab Instructor Use ONLY--------------------------

GRADE: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

COMMENTS: 

Electrical and Computer Engineering Department

School of Engineering

Rutgers University, Piscataway, NJ 08854

In**troduction:** GPU uses a different ISA than MIPS. The ISA used is PTX ISA, which is more efficient for GPU’s

**Assignment 1:**

The program loads integers from the local memory. It also loads the array e. Since the array data is in order in the memory. The head of the array is loaded and then the offset is changed when going through the array items. The predicate p is set by the greater than instruction to set where the loop starts and ends. Other predicate is used for the if condition statement and is used inside the other loop. When @!p happens, that means the for loop is over.

**Assignment 2:**

The registers that are 64 bits take up the space of 2 32 bit registers. Therefore, it takes the space of two registers but only has to be implemented once. The next register would start from after the end of the 64 bits of the previous register. The floating point type is used so that division can be used and the decimal values will be saved.

**Assignment 3:**

In the MIPS version of the geometric series, there needs to be more instructions to hold values such as the exponent 2^k and the current sum. Also, a makeshift binary exponent function had to be created in the MIPS version. However, in the PTX program, there is a binary exponent instruction. Therefore, each additional iteration of n causes the binary exponent function to be used again. So the instructions saved with the binary exponent instruction is multiplied to the amount of times it is called in the series.

The PTX saved 4 instructions using the binary exponent instruction compared to the MIPS version. If there are 10 cycles of the loop, then 4\*10=40 instructions is saved.

**Assignment 4:**

The bubble sort has two loops. One outer loop that keeps track of how many times the array has been traversed already. The inner loop contains the traversal. Each loop iteration does a comparison between the current index in the array and the next index. The inner loop goes until the 2nd to last index because at this point, the comparison includes the last index already.

**Conclusions and discussions:**

The PTX version of the code is much shorter because there are more efficient instructions and syntax. The MIPS version has more temporary registers to hold values as more instructions are needed to do the same thing as in PTX. In addition, predicates are used in PTX ISA which saves instructions and makes the code easier to follow.