

**CSCE 313**  
**Lab 3**  
**Performance Analysis**  
**Due Date: 3/4, 11:59pm**

**Design Requirements**

This lab's objective is to use the debugger and performance counters to measure the performance of various aspects of your code from Lab 2.

First:

1. add support for hardware floating-point instructions
2. add a performance counter
3. delete the interval timer

Measure how many cycles are required on average to transform one pixel with the rotation transform using the following cache configurations for the NIOS II/f:

1. 4K instruction cache, 4K data cache
2. 4K instruction cache, 16K data cache

After this, replace all **floating point** multiplies and adds with **fixed point** multiplies and adds. Maintain an accuracy of at least  $2^{-9}$ . Repeat the same tests as above.

Use preprocessor directives such as `#define` and `#ifdef` to make it convenient to switch between fixed point and floating point. For example:

```
#define      FIXED_POINT

...

#ifdef FIXED_POINT

sin_val = (alt_32)(sinf(angle) * 512.0);
cos_val = (alt_32)(cosf(angle) * 512.0);

#else

sin_val = sinf(angle);
cos_val = cosf(angle);

#endif
```

**Project Submission**

Each group must submit a report that includes:

- Instructions per pixel for floating-point and fixed-point
- Cycles per pixel for floating-point and fixed-point each with 4K and 16K data cache size