Instruction variance analysis

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

We can divide the Sass code into 5 stage:

1: Start;

2: Compute\_ID;

3: Load from memory;

4: Do Addition;

5: End;

**Instruction counts**

**#compute\_id**: instructions used to compute thread id, memory placement dependent.

**#data\_objects**: data objects that involved with memory placement.

in this case, #data\_objects = 2 [only array a and b]

**#Static**: static instructions that independent with memory placement

in this case, #Static = stage 1+ stage4 + stage 5 = 1+4+1=6

**#memory\_load** : instructions that related to memory load.

global memory = 3

constant memory = 1

texture memory = 1

share memory = 5

#Total Instructions = #Static + #compute\_id + #data\_objects \* #meomry\_load

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stage | vd | vd\_constant | Vd\_tex | Vd\_shared |
| 1 | 1 | 1 | 1 | 1 |
| 2 | 4 | 5 | 4 | 5 |
| 3 | 6 | 2 | 2 | 10 |
| 4 | 4 | 4 | 4 | 4 |
| 5 | 1 | 1 | 1 | 1 |
| Total instructions | 16 | 13 | 12 | 21 |