HIGH PERFORMANCE COMPUTER ARCHITECTURE *final-term exam 21-12-2016* **MATR.NO.**_____

(REVISION 1.2) **SURNAME**_____

FIRST NAME_____

1) (POINTS 10/20) Consider a bus-based multicore that support the DRAGON cache coherence protocol. The cost of a read/write operation is 1 cycle, the cost of a BusRd (or BusRdX) transaction is 90 cycles; all caches are write-back, write-allocate and initially empty. For the DRAGON protocol the BusUpgr is not used. The cost of a BusUpdate is 60 cycles. Evaluate the total cost of executing the following streams by completing the table below:

Stream1: R1, W1, R1, W1, R2, W2, R2, W2, R3, W3, R3, W3
Stream2: R1, R2, R3, W1, W2, W3, R1, R2, R3, W3, W1
Stream3: R1, R2, R3, R3, W1, W1, W1, W1, W2, W3

stream-1 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1						
PrWr1						
PrRd1						
PrWr1						
PrRd2						
PrWr2						
PrRd2						
PrWr2						
PrRd3						
PrWr3						
PrRd3						
PrWr3						
					TOTAL	

stream-2 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1						
PrRd2						
PrRd3						
PrWr1						
PrWr2						
PrWr3						
PrRd1						
PrRd2						
PrRd3						
PrWr3						
PrWr1						
					TOTAL	

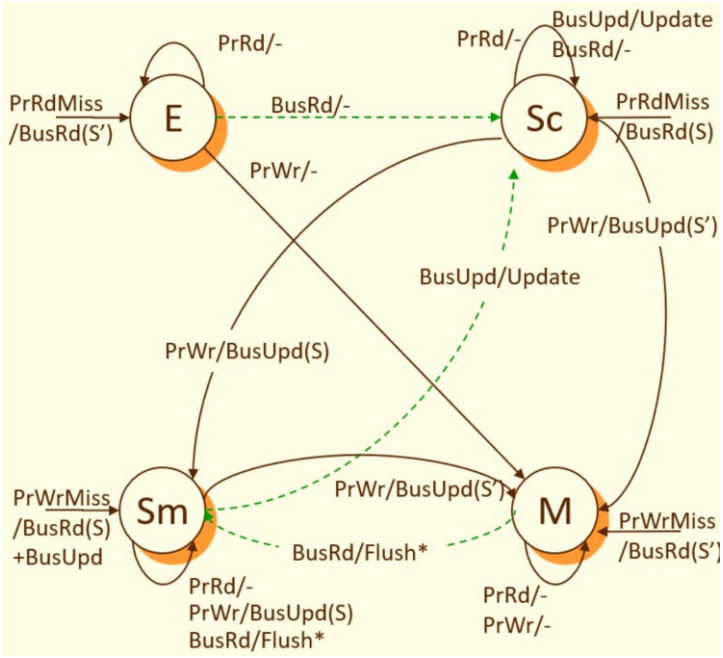
stream-3 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1						
PrRd2						
PrRd3						
PrRd3						
PrWr1						
PrWr1						
PrWr1						
PrWr1						
PrWr2						
PrWr3						
					TOTAL	

2) (10/20) For the following CUDA kernel, check if it will cause branch divergence for warp 0 and for warp 1 in thread block 0 assuming that each thread block contains more than 2 warps (explain in detail the result).

```
gid= threadIdx.x + blockIdx.x * blockDim.x;
If (gid < 8) { //this the branch of interest
    //TAKEN_PATH
} else {
    //NOT_TAKEN_PATH
}
```

SOLUTION EXERCIZE 1:



NOTE1:
We do not account for cycle to address the cache or the memory we just account for the time to bring the data from cache into processor (e.g., 1 cycle) or from memory to cache (e.g., 90 cycles)

NOTE2:
We use Flush* since we only update blocks among caches; the memory is updated by the cache with the Sm block when the Sm block is evicted

stream-1 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1	E			BusRd(S')	Mem	90
PrWr1	M					1
PrRd1	M					1
PrWr1	M					1
PrRd2	Sm	Sc		BusRd(S)/Flush*	P1.C	90
PrWr2	Sc	Sm		BusUpd(S)		60
PrRd2	Sc	Sm				1
PrWr2	Sc	Sm		BusUpd(S)		60
PrRd3	Sc	Sm	Sc	BusRd(S)/Flush*	P2.C	90
PrWr3	Sc	Sc	Sm	BusUpd(S)		60
PrRd3	Sc	Sc	Sm			1
PrWr3	Sc	Sc	Sm	BusUpd(S)		60
					TOTAL	515

stream-2 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1	E			BusRd(S')	Mem	90
PrRd2	Sc	Sc		BusRd(S)	Mem	90
PrRd3	Sc	Sc	Sc	BusRd(S)	Mem	90
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
PrWr2	Sc	Sm	Sc	BusUpd(S)		60
PrWr3	Sc	Sc	Sm	BusUpd(S)		60
PrRd1	Sc	Sc	Sm			1
PrRd2	Sc	Sc	Sm			1
PrRd3	Sc	Sc	Sm			1
PrWr3	Sc	Sc	Sm	BusUpd(S)		60
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
					TOTAL	573

stream-3 DRAGON

Core Operation	C1	C2	C3	Bus Transaction	Data from	Cycles
PrRd1	E			BusRd(S')	Mem	90
PrRd2	Sc	Sc		BusRd(S)	Mem	90
PrRd3	Sc	Sc	Sc	BusRd(S)	Mem	90
PrRd3	Sc	Sc	Sc			1
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
PrWr1	Sm	Sc	Sc	BusUpd(S)		60
PrWr2	Sc	Sm	Sc	BusUpd(S)		60
PrWr3	Sc	Sc	Sm	BusUpd(S)		60
					TOTAL	631

SOLUTION EXERCIZE 2:

The code will cause divergence for warp 0 since the first 8 threads out of 32 in the first block will take the branch while the other 24 no.

In the case of warp 1 (same block) all the threads will not take the branch so there is no divergence.