Exam I

ENGR E315 / ENGR E599 / CS B649

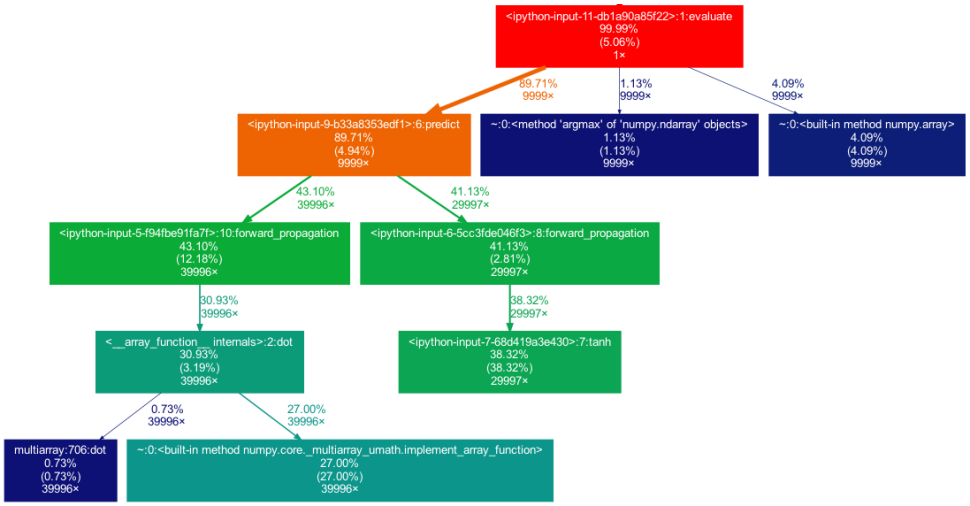
Name: _____KEY_____

Email: _____

--------- This page intentionally left blank. ---------

1. Performance Profiling (20 Points)

```
ncalls  tottime  percall  cumtime  percall filename:lineno(function)
 39996    0.028    0.000    0.264    0.000 <__array_function__ internals>:2(dot)
     1    0.043    0.043    0.849    0.849 <ipython-input-11-db1a90a85f22>:1(evaluate)
 39996    0.099    0.000    0.364    0.000 <ipython-input-5-f94fbe91fa7f>:10(forward_propagation)
 29997    0.022    0.000    0.355    0.000 <ipython-input-6-5cc3fde046f3>:8(forward_propagation)
 29997    0.333    0.000    0.333    0.000 <ipython-input-7-68d419a3e430>:7(tanh)
  9999    0.041    0.000    0.764    0.000 <ipython-input-9-b33a8353edf1>:6(predict)
     1    0.000    0.000    0.849    0.849 <string>:1(<module>)
 39996    0.007    0.000    0.007    0.000 multiarray.py:706(dot)
     1    0.000    0.000    0.849    0.849 {built-in method builtins.exec}
 10000    0.002    0.000    0.002    0.000 {built-in method builtins.len}
  9999    0.032    0.000    0.032    0.000 {built-in method numpy.array}
 39996    0.230    0.000    0.230    0.000 {built-in method numpy.core._multiarray_umath.implement_array_function}
  9999    0.002    0.000    0.002    0.000 {method 'append' of 'list' objects}
  9999    0.010    0.000    0.010    0.000 {method 'argmax' of 'numpy.ndarray' objects}
     1    0.000    0.000    0.000    0.000 {method 'disable' of '_lsprof.Profiler' objects}
```



a) Given the performance profiling information above, which two functions would you focus on for acceleration attempts?

(5 pts) tanh (38.32%) and Numpy…implement_array_function (27.00%). Dot (30.93% with subfunctions) also accepted.

b) Why did you choose those two functions?

(7.5 pts / ea)

Tanh: This function requires 38.32% of the total execution time. This is the largest percentage for a single function which does not call sub-functions

Numpy's implement_array_function: This function requires 27.00% of the total time. This is the second largest percentage for a single function which does not call sub-functions.

Dot: This function requires 30.93% of the total execution time, but calls a numpy routine. Focusing on reducing the calls to numpy may accelerate this function's computations.

2. Data Structures (10 Points)

Please circle which data structure is expected to offer better performance for the given task. (2pts/ea)

a. Inserting new items at the beginning?

Linked List Array

b. Accessing arbitrary elements?

Linked List Array

c. Accessing elements sequentially?

Linked List Array

d. Being copied?

Linked List Array

e. Using the least memory?

Linked List Array

3. MultiThreading vs. MultiProcessing (10 Points)

a) What is the difference between Multithreading and Multiprocessing?

(4 pts) Multithreading creates multiple threads within a single process. These threads have separate registers and stacks, but all share a single heap. Multiprocessing creates multiple processes, each with a single thread. These threads have separate registers, stacks, and heaps. Multithreading is faster to create/maintain, but multiprocessing allows more freedom between threads.

b) When is multithreading useful in Python?

(3 pts) When one or more thread is blocked on Input/Output, it will release the Global Interpretter Lock (GIL), allowing other threads to run.

c) When is multiprocessing useful in Python?

(3 pts) When multiple threads want to execute in parallel, Python's GIL will serialize them. Multiprocessing enables multiple Python processes to run in parallel.

4. Bus Interfaces (40 Points)

a. What is the difference between a bus master and bus slave?

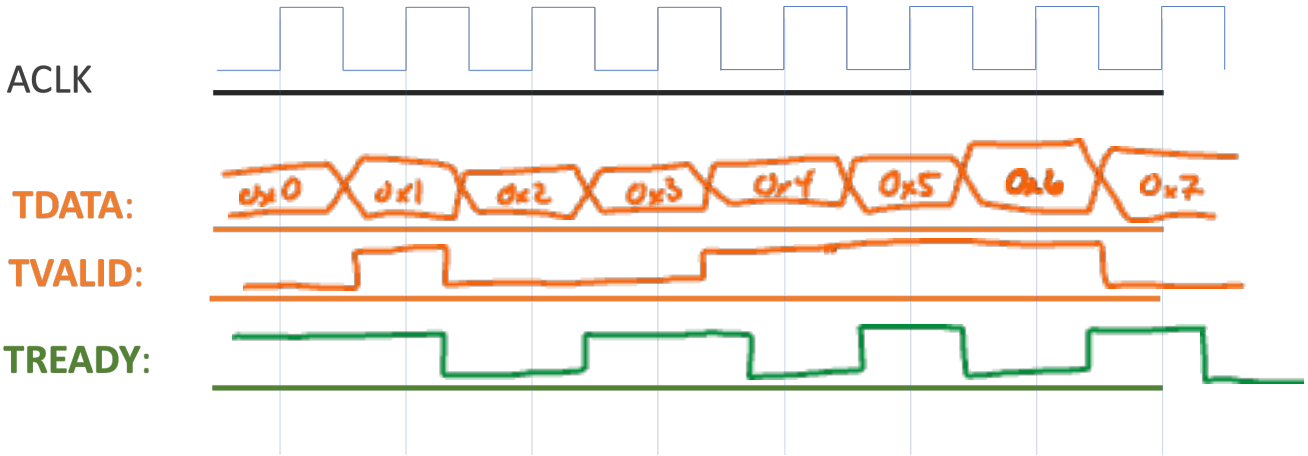
(3 pts) Master can initiate transactions, slave only responds

b. Is AXI4 Stream Memory-Mapped? (Circle one) (2pts)

Yes No

c. Explain AXI Stream handshake? When is transmission valid?
(6 pts) A standardized interface for transmitting data between a master and slave. Transmissions are valid when both the master's TVALID and the slave's TREADY are high at the positive (rising) edge of ACLK.

d. What values are transmitted on this AXI4 Stream Bus?



(4pts/ea) 0x1, 0x5

e. For AXI4 Lite, what is one example of a master and a slave? (other than the FPGA) (2 pts/ea)

i. Master: CPU (DMA)
ii. Slave: Memory (DMA)

f. What is the purpose of a bus arbitrator? (4 pts)

Select which bus master gets to initiate transactions if multiple masters request simultaneously (contention)

g. What is the different in the Round Robin and Fixed Priority bus arbitration schemes? (5 pts)

With Fixed Priority, the priority of the bus masters is fixed. One bus master always has priority over the other(s). With Round Robin, the priority of the bus masters rotates. Thus each bus master only has priority over others, but only for a short time.

h. What is one benefit and one downside of using AXI4 Full vs. AXI4 Lite? (2 pts)

AXI4 Full is faster but more complex than AXI4 Lite

i. What does the acronym "MMIO" mean? (2 pts)

Memory-Mapped Input/Output

j. Using MMIO, how does the CPU interact with peripherals and the FPGA? (4 pts)

Regular loads and stores to memory addresses

DMA (20 pts)

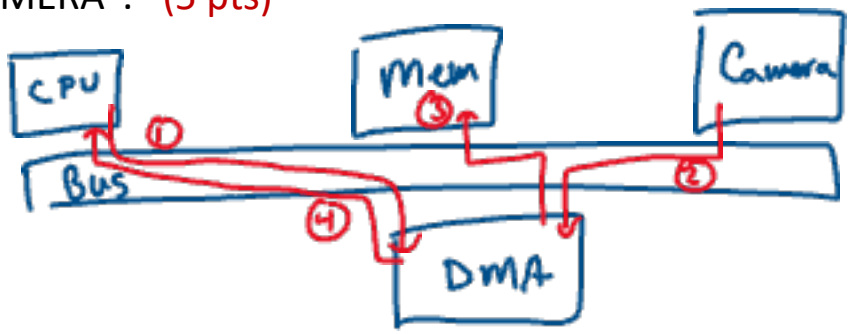
a. What does the "Direct" in "DMA" mean? (2 pts)

The DMA is a bust master that can initiate transactions without direct CPU involvement.

b. Why do DMA units have both a master and slave memory interface? (3 pts)

Master: Initiate bus transactions to copy memory
Slave: Respond to MMIO requests from CPU to control DMA

c. Draw the data movement used by the CPU to copy data the Camera to the memory using DMA. Your picture should include the blocks "BUS", "CPU", "DMA", "MEM" and "CAMERA". (5 pts)



d. In a few sentences, describe the sequence of bus transactions that occur in your diagram above. (10 pts)

1. CPU starts transaction (2 pts)
2. DMA loads from Camera (2 pts)
3. DMA sends to Mem (2 pts)
4. Steps 2+3 repeat until all data copied (2 pts)
4. CPU polls until DMA complete (2 pt)