CST1
COMPUTER SCIENCE TRIPOS Part IB

Tuesday 6 June 2023   13:30 to 16:30

COMPUTER SCIENCE Paper 5

Answer *five* questions.

Submit the answers in five **separate** bundles, each with its own cover sheet. On each cover sheet, write the numbers of **all** attempted questions, and circle the number of the question attached.

> **You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator**

STATIONERY REQUIREMENTS
Script paper
Blue cover sheets
Tags

SPECIAL REQUIREMENTS
Approved calculator permitted

1 Computer Networking

Recall the simplified TCP throughput equation

$$\text{TCP throughput} = \frac{1.22 \cdot \text{MSS}}{\text{RTT} \cdot \sqrt{L}}$$

(a) Provide a derivation of this equation utilising a figure. Include a description of each term of this equation and example values (including units).   [6 marks]

(b) What does this equation imply for networks of $10\,\text{Gbit/s}$ throughput.

[3 marks]

(c) What important TCP congestion behaviour does this equation not capture?

[3 marks]

(d) CUBIC is commonly used as an alternative to classic AIMD TCP congestion control.

(i) With the aid of a diagram showing window size over time, compare how CUBIC differs from classic AIMD TCP congestion control.   [6 marks]

(ii) With reference to the diagram used in part $(d)(i)$ discuss how the CUBIC approach improves performance for a flow on a link with very large bandwidth delay products.   [2 marks]

2 Computer Networking

ISP ShinyNet connects customers to the Internet with a symmetric 40 Mbit/s link. The ISP gateway router/modem has a single buffer shared by all senders with a capacity of 1000 packets. You may assume all packets are of the same length and that packet-arrivals are not bursty.

A popular application uses three UDP flows each concurrently sharing the link. The three flows send at rates of 10 Mbit/s, 20 Mbit/s and 30 Mbit/s respectively.

(*a*) What is the data rate successfully traversing the buffer and the fraction of packets for each flow dropped by the buffer? [3 marks]

ShinyNet upgrades the customer gateway router/modem. The upgraded device employs one queue per flow (three queues in total for this case); each queue can holds a maximum number of packets in proportion to the flow count (333 packets per flow), the queues are scheduled using simple per-flow fair queueing with an equal weight for each queue.

(*b*) What is the packet loss rate for each flow when using the upgraded gateway? [3 marks]

(*c*) Approximately how many packets does each flow have in its queue? [3 marks]

ShinyNet updates the gateway routers/switch to now use weighted-fair queueing. Using an unknown patented technology (or poor configuration); the following weights are assigned to the flows as follows: 0.2, 0.6, and 0.2 for each flow respectively.

(*d*) What is the packet loss rate for each flow when using this gateway with weighted fair-queueing? [3 marks]

(*e*) Approximately how many packets does each flow have in its queue? [3 marks]

(*f*) Throughout this question we have assumed packets are of equal length; this is not a valid assumption for the vast majority of Internet traffic. Presuming that over the long term we wish for the queue discipline to retain the desired weighted-fairness properties, discuss the issues variable length packets raise, and propose an approach to maintain the desired weighted-fairness outcome. [5 marks]

3 Computer Networking

(*a*) (*i*) Ethernet switches use a spanning tree. Explain briefly what the aims of the spanning tree protocol are, and how it works to achieve these aims.

[6 marks]

(*ii*) Do the switches learn the entire network topology? Explain your answer.

[2 marks]

(*iii*) Routers using link-state protocols communicate over the shortest path. Does each pair of switches communicate over the shortest path?

[2 marks]

(*iv*) Routers may use a link-state protocol or a distance vector protocol. Compare the message-size complexity, computational complexity, and the robustness of these two approaches.　　　　　　　[6 marks]

(*b*) The computer of another student on your college stairwell doesn't connect to the Internet.

They go on to say "...it's weird — my computer can talk with the printer in my room and my computer can see your computer, but I can't upload this essay due tonight, I can't connect to Google, and I can't even send an email..."

You suspect their computer is using automatically-allocated *link-local* addresses for both IPv4 and IPv6.

Speculate what has gone wrong to cause the computers to be using link-local addresses. You may want to consider: What address did the machine use? Why is the computer using link-local addresses? Why are some services are working but other services are not?

[4 marks]

4 Concurrent and Distributed Systems

(a) A guarded resource needs to be locked in three different ways by readers, moderators and writers. A thread will gain access to the resource in one of these three ways, operate, and then relinquish access. At any instant, the resource may be unlocked or held be any number of readers, or up to two moderators, or at most one writer.

(i) Define some number of locks, mutexes and/or shared variables to manage the system. Sketch the core of a state transition diagram.   [5 marks]

(ii) Using a monitor approach, give pseudocode for these six methods:

```
start_write()   start_moderate()   start_read()
end_write()     end_moderate()     end_read()
```
    [5 marks]

(b) Very briefly describe two techniques for deadlock avoidance and one technique for graceful deadlock recovery (i.e. not reboot). Describe two burdensome aspects of deadlock recovery.   [6 marks]

(c) A system has T threads. These use 3 types of resource that each have 4 instances provided. A deadlock avoidance system restricts resource acquisition. What state space does the system potentially have before and after restriction?
    [4 marks]

(TURN OVER)

5 Concurrent and Distributed Systems

A system is being designed to measure traffic flow into a city. Part of this system is a set of monitoring nodes, each using sensors to detect when a vehicle passes the monitoring node, and incrementing a per-node counter. The nodes communicate over a network to provide a city-wide total.

(*a*) Define each of the terms:

   (*i*)   Fair-loss network links.   [1 mark]

   (*ii*) Crash-recovery execution.   [1 mark]

   (*iii*) Asynchronous timing.   [1 mark]

(*b*) Consider a version of the system using *quorum-based replication.* There is a fixed set of 5 nodes, meaning that each node holds a 5-tuple comprising the node's most recent values for each of the replicated counters. A node should be able to operate if it can communicate with at least 2 other nodes. Describe how each of the following operations can be implemented by sending messages between nodes:

   (*i*)   A `setCount(n)` function to update the node's local counter to `n` and to replicate the change to a quorum.   [5 marks]

   (*ii*) A `getTotalCount()` function to return the total of the counters from all of the nodes.   [5 marks]

   You should describe the messages sent and received by each node, along with how a node updates its local 5-tuple with new information when it receives messages, and how a node determines that the operation is complete.

(*c*) Does your system provide *linearizable* behaviour? Either explain why your system is linearizable, or provide an example showing a non-linearizable result.
                                                                            [3 marks]

(*d*) Consider a second version of the system that provides *strong eventual consistency* and allows the operations to always complete irrespective of the number of nodes available for communication. Summarize the changes needed to provide this behaviour.   [4 marks]

6 Introduction to Computer Architecture

Consider the following three correct state machines (`seqA`, `seqB`, `seqC`) written in SystemVerilog.

```
module seqA(input clk, input rst,          module seqB(input clk, input rst,
            output logic [3:0] a);                     output logic [3:0] b);
                                           logic [3:0] n;
always_ff @(posedge clk or posedge rst)    always_ff @(posedge clk or posedge rst)
  if(rst)                                    if(rst) b <= 0;
    a <= 0;                                  else    b <= n;
  else                                     always_comb
    begin                                    begin
      a[0] <= !a[0];                           n[0] = !b[0];
      a[1] <=  a[0]    ^ a[1];                 n[1] = (b[0] & !n[0]) ^ b[1];
      a[2] <= &a[1:0] ^ a[2];                  n[2] = (b[1] & !n[1]) ^ b[2];
      a[3] <= &a[2:0] ^ a[3];                  n[3] = (b[2] & !n[2]) ^ b[3];
    end                                      end
endmodule                                  endmodule
```

```
module seqC(input clk, input rst, output logic [3:0] c);
logic [15:0] s;
always_ff @(posedge clk or posedge rst)
  if(rst) s <= 16'd1;
  else    s <= {s[14:0],s[15]};
always_comb
  begin
    c[0] = s[1] | s[3] | s[5] | s[7] |  s[9] | s[11] | s[13] | s[15];
    c[1] = s[2] | s[3] | s[6] | s[7] | s[10] | s[11] | s[14] | s[15];
    c[2] = (|s[7:4]) | (|s[15:12]);
    c[3] = |s[15:8];
  end
endmodule
```

(a) Why is it important that an implementation of a circuit meets all timing constraints? [2 marks]

(b) What is the complete sequence that each of the three modules (`seqA`, `seqB`, `seqC`) outputs after reset (`rst`) is released? Justify your answer. [6 marks]

(c) If the three modules were mapped to an FPGA consisting of many 4-input 1-output LUTs (lookup tables), DFFs (D flip-flops) and programmable wiring, what resources would each module require for a minimal implementation? Justify your answer. [6 marks]

(d) Let us assume that LUTs have an input-to-output delay of $2d$, DFFs have a setup time of d and no other delays, and we ignore wire delays. For each module, what is the minimum clock period in terms of d assuming no clock jitter? Justify your answer. [6 marks]

7 Introduction to Computer Architecture

(a) What is Amdahl's law? [2 marks]

(b) For a 64-bit RISC-V pipelined processor, how could the following code be optimised to reduce execution time? Justify your answer. [4 marks]

```
lw t0,  0(a0)   # load 32-bits from address a0 into register t0
sw t0,  0(a1)   # store 32-bits to address a0 from register t0
lw t1,  4(a0)
sw t1,  4(a1)
lw t2,  8(a0)
sw t2,  8(a1)
lw t3, 12(a0)
sw t3, 12(a1)
```

(c) A new RISC-V R-type instruction `swap` is proposed to swap two registers. For example:

```
swap t0, t1
```

would swap the contents of registers `t0` and `t1`. What would be the challenges in implementing this proposed `swap` instruction? [3 marks]

(d) A counterproposal is to introduce `swap` as a pseudo instruction that unpacks to a short sequence of real instructions. What sequence of instructions could be generated that swaps two registers without using a additional register? [*Hint: it involves using `xor`*] [3 marks]

(e) RISC-V provides an atomic swap in memory operation `amoswapd` that takes an address held in a register (e.g. `a0`), a source register (e.g. `t0`), a destination register (e.g. `t1`). For these example register allocations, the operation performs an atomic read-modify-write where the value at address `a0` is stored in register `t1` and the value in register `t0` is written to address `a0`. How could the same operation be achieved using load-reserved (`lr`) and store-conditional (`sc`) instructions instead of using `amoswapd`? Explain your code by commenting it. [4 marks]

(f) If a computer system uses the MSI cache coherence protocol, and if multiple processor cores are reading shared data at address a at the same time, what happens when one processor core writes a word to address a? [4 marks]

8 Introduction to Computer Architecture

(a) What is the *power wall* and why does it lead to the idea of *dark silicon*?

[3 marks]

(b) What is the difference between race-to-dark (also known as race-to-sleep or race-to-halt) and dynamic voltage/frequency scaling in order to save power?

[3 marks]

(c) Why does dark silicon lead system-on-chip designers to incorporate accelerators? Give an example of an accelerator to illustrate your answer.   [4 marks]

(d) Why might systems-on-chip contain many more processor cores than the application class cores, which are typically the ones advertised? What characteristics do these cores possess?   [4 marks]

(e) Why do you need to perform both a row access and column read when reading data out of DRAM? What does each operation do?   [3 marks]

(f) DRAM can perform burst reads and writes of several words. How does the last-level cache use and benefit from DRAM burst accesses?   [3 marks]

END OF PAPER