CST2
COMPUTER SCIENCE TRIPOS Part II

Monday 9 June 2025 13:30 to 16:30

COMPUTER SCIENCE Paper 8

*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 Advanced Computer Architecture

(*a*) Why might it be reasonable to estimate the performance of a high-performance processor as being proportional to the square root of the area of the core? [4 marks]

(*b*) It has become increasingly difficult to boost the performance of high-performance superscalar processors. This has raised concerns within industry that only very modest gains might be possible in the future.

(*i*) Should we be concerned about the stagnation of single-thread performance? Provide one argument in favour and one against. [6 marks]

(*ii*) Some suggest that industry should explore radically different processor designs to boost single-thread performance. Why have developments in Instruction Set Architectures (ISAs) and microarchitectures historically been more incremental in nature? [3 marks]

(*iii*) One potential avenue for innovation involves enhancing the information shared between the compiler and hardware. This would be achieved by modifying the Instruction Set Architecture (ISA). What are the possible benefits of this approach, and what challenges might arise in practice? [4 marks]

(*c*) A chiplet is a small, modular integrated circuit that can be combined with other chiplets to create a complex system within a single package. What are the benefits of this approach when compared to designing and manufacturing a single die? [3 marks]

2 Bioinformatics

(*a*) In genome alignment we need to detect *inexact* matching with respect to the reference genome in order to find variants. This allows characterising individual differences and disease-causing mutations. Briefly explain, with one example, what the Burrows–Wheeler Transform (BWT) is. State its advantage over naive search algorithms when finding exact genome alignments. Explain how it may be extended to detect variants. [8 marks]

(*b*) The polymerase chain reaction is used to amplify RNA sequences. There are short sequences, primers, that start the alignment. These should not form extensive pairings (for example, in stem-loop structures).

Discuss with one example (4 or 5 bases) how Ruth Nussinov's RNA algorithm can indicate whether an RNA sequence may be suitable. [7 marks]

(*c*) Experimental evidence shows that two proteins A and B bind each other.

(*i*) If experiments show that the two proteins interact in many species, should we expect that they have similar phylogenetic trees when you use the A and B sequences from these species? [3 marks]

(*ii*) Discuss the role of a scoring matrix in optimal tree phylogeny. [2 marks]

3 Cryptography

LegacyGates Ltd has for decades sold garage-door openers that implement a simple challenge–response authentication protocol using their time-honoured 42-bit blockcipher E:

Door D:		Key fob F:
pick $C \in_R \{0,1\}^{42}$	$\xrightarrow{\quad C \quad}$	
		$R := E_K(C) \in \{0,1\}^{42}$
open if $R = E_K(C)$	$\xleftarrow{\quad R \quad}$	

$K \in \{0,1\}^{42}$ is a private key shared between door D and radio key fob F.

E and E^{-1} have become publicly known and burglars have started to eavesdrop some challenge–response pairs (C_1, R_1), (C_2, R_2), (C_3, R_3) and brute-force search for a K that satisfies $R_i = E_K(C_i)$. Renting servers that can try 2^{42} encryptions per day, or even store 2^{42} pairs of blocks in a lookup table, is now quite affordable.

Product teams have been asked to upgrade the protocol to 84-bit security. For "commercial reasons" the company wants to continue to use E, but with a pair of 42-bit keys (K_1, K_2). Five teams each came up with a different proposal. Your task is to evaluate if they succeeded.

For each of the following modifications of the protocol, estimate the number of test encryptions needed, and the amount of storage required, to find a pair (K_1, K_2) that will open the door, and give the corresponding search algorithm.

(*a*) $R := E_{E_{K_2}(K_1)}(C)$ [3 marks]

(*b*) $R := E_{K_1}(C) \oplus K_2$ [4 marks]

(*c*) $R := E_{K_2}(E_{K_1}(C))$ [4 marks]

(*d*) $R := E_{K_2}(E_{K_2}(E_{K_1}(C)))$ [4 marks]

(*e*) $R := E_{K_1}(E_{K_2}(E_{K_1}(C)))$ [4 marks]

Regarding proposal (*e*):

(*f*) What practical advantage would using $R := E_{K_1}(E_{K_2}^{-1}(E_{K_1}(C)))$ instead have? [1 mark]

4 Denotational Semantics

In your answers, you are allowed to use theorems from the course, provided you state them precisely beforehand.

Let $(P, \sqsubseteq)$ be a poset. We say a subset $S \subseteq P$ is

- a downset if whenever $y \in S$ and $x \sqsubseteq y$ then also $x \in S$;
- chain-closed if for any chain $x_0 \sqsubseteq x_1 \sqsubseteq \cdots \in S$, $\bigsqcup_i x_i \in S$ whenever the lub exists (in P).

We write $\mathcal{D}(P)$ (respectively $\mathcal{C}(P)$) for the set of downsets (resp. chain-closed downsets) of P, and $\mathcal{P}(S)$ for the powerset of a set S. Given a function $f \in X \to Y$, we write $f^{-1} \in \mathcal{P}(Y) \to \mathcal{P}(X)$ for the inverse image function, which maps a subset $S \subseteq Y$ to $f^{-1}(S) = \{x \in X \mid f(x) \in S\}$.

(*a*) Show that for any set X, $(\mathcal{P}(X), \subseteq)$ is a domain. [4 marks]

(*b*) Show that given any two sets X and Y and a function $f \in X \to Y$, f^{-1} is a strict continuous function. [4 marks]

(*c*) Given two posets P and Q, show that if a function $f \in P \to Q$ is monotone then for all downsets $D \in \mathcal{D}(Q)$, $f^{-1}(D)$ is a downset. [3 marks]

(*d*) Show the converse: if P and Q are two posets, and $f \in P \to Q$ a function such that f^{-1} maps downsets to downsets, then f is monotone. [*Hint:* You might want to consider $\downarrow a$, the set $\{x | x \sqsubseteq a\}$ of elements smaller than a.] [4 marks]

(*e*) Given two chain complete partial orders P and Q, show that a monotone function $f \in P \to Q$ is continuous if and only if f^{-1} maps chain-closed downsets to chain-closed downsets. [5 marks]

5 E-Commerce

(*a*) Many software based businesses take advantage of information economics. Use an example to describe why the information economics of many software markets give rise to dominant firms. [5 marks]

(*b*) Businesses are currently exploring the best delivery model for the training and use of large language-model based chatbots to users. With reference to economic, legal and product concerns make a recommendation to a product developer, discussing the advantages and disadvantages of using a software-as-a-service (SaaS) web app or a client-side dedicated hardware appliance, to deliver a large language model chatbot to a user. [15 marks]

6 Hoare Logic and Model Checking

Consider a programming language with commands C consisting of the `skip` no-op command, sequential composition C_1`;`C_2, loops `while` B `do` C for Boolean expressions B, conditionals `if` B `then` C_1 `else` C_2, assignment X `:=` E for program variables X and arithmetic expressions E, heap allocation X `:= alloc(`E_1`,...,`E_n`)`, heap assignment `[`E_1`]` `:=` E_2, heap dereference X `:= [`E`]`, and heap location disposal `dispose(`E`)`. Assume `null` $= 0$, and predicates for lists and partial lists:

$$\begin{aligned}
&\text{list}(t, []) = (t = \text{null}) \wedge emp \\
&\text{list}(t, h :: \alpha) = \exists y.(t \mapsto h) * ((t+1) \mapsto y) * \text{list}(y, \alpha) \\
&\text{plist}(t_1, [], t_2) = (t_1 = t_2) \wedge emp \\
&\text{plist}(t_1, h :: \alpha, t_2) = \exists y.\ (t_1 \mapsto h) * ((t_1+1) \mapsto y) * \text{plist}(y, \alpha, t_2)
\end{aligned}$$

In the following, all triples are linear separation logic triples.

(a) Give a proof outline for the following triple or explain why a proof must fail.
$\{P = p \wedge p \mapsto v\}$ `Q:=P; dispose(Q); [P]:=1;` $\{p \mapsto 1\}$ [2 marks]

(b) Define a separation logic predicate array(t, α) for pointers t and mathematical lists α that, unlike the list predicate, represents the values of α in consecutive memory cells starting from t. [3 marks]

(c) Define and explain a partial correctness rule for a new command `nondet`. For a given command C, `nondet`(C) nondeterministically either executes C or does nothing. Maintain soundness of the proof system, and ensure the rule accurately reflects the behaviour of the new command. [3 marks]

(d) Give a loop invariant that would prove the following triple, for a command that interleaves two lists of the same length (no proof required). The zip function interleaves mathematical lists, for example zip $[1,3,5]$ $[2,4,6] = [1,2,3,4,5,6]$.
$\{(\text{list}(X, \alpha) * \text{list}(Y, \beta)) \wedge \text{length}\ \alpha = \text{length}\ \beta\}$
```
P:=X; Q:=Y;
while P ≠ null do
  (N1:=[P+1]; N2:=[Q+1]; [P+1]:=Q; [Q+1]:=N1; P:=N1; Q:=N2)
```
$\{\text{list}(X, \text{zip}\ \alpha\ \beta)\}$ [5 marks]

(e) Give a loop invariant that would prove the following triple, for a command that splits a non-empty list of length $2n$ (for some n) into two lists of alternating elements (no proof required).
$\{\text{list}(X, \alpha) \wedge \alpha \neq [] \wedge \exists n.\ \text{length}\ \alpha = 2n\}$
```
Y:=[X+1]; LX:=X; LY:=Y; N:=[LY+1];
while N ≠ null do
  ([LX+1]:=N; LX:=N; N:=[N+1]; [LY+1]:=N; LY:=N; N:=[N+1]);
[LX+1]:=null; [LY+1]:=null;
```
$\{\exists \beta, \gamma.\ (\text{list}(X, \beta) * \text{list}(Y, \gamma)) \wedge \alpha = \text{zip}\ \beta\ \gamma\}$ [7 marks]

7 Information Theory

(a) For two random variables, X and Y, draw a diagram to depict the relationship between the individual entropies, the joint entropy, the conditional entropies and the mutual information. [2 marks]

(b) You have an unbiased coin. You flip it once. If it is heads, you flip once more. If it is tails you flip twice more. You do not reveal the outcomes, only the total number of heads.

(i) By modelling this as a channel, compute how much information about the outcome of the first flip someone can get from knowing only the total number of heads that occurred. [7 marks]

(ii) Provide an intuitive explanation of your answer to (b)(i). [2 marks]

(c) A channel has an input alphabet of $X = \{0, 1, 2\}$ with input probabilities $\{p_0, p_1, p_2\}$, respectively, and an output alphabet of $Y = \{0, 1, 2\}$.

(i) Derive an expression for the capacity of the channel if the transition probabilities satisfy:

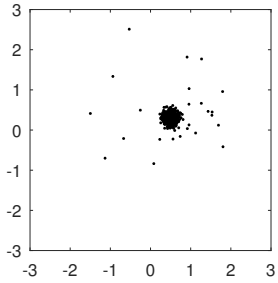
$$\begin{aligned} P(Y = i | X = i) &= a \\ P(Y = (i+1) \bmod 3, X = i) &= b \\ P(Y = (i+2) \bmod 3, X = i) &= c \end{aligned}$$

[6 marks]

(ii) Assuming $b = c = \frac{(1-a)}{2}$, compute the channel capacity as a approaches $\frac{1}{3}$ and 1. Give an intuitive explanation. [3 marks]

8 Machine Learning and Bayesian Inference

Evil Robot is shopping for Death Rays. He attends a presentation of two Death Rays, but is distracted and misses the demonstration, after which the target looks as follows:



He concludes that one is fast and accurate while the other is slower and inaccurate. To estimate the accuracy and speed of each, he models hits on the target as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \pi\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \sigma_1\mathbf{I}) + (1-\pi)\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \sigma_2\mathbf{I})$$

where $\mathcal{N}$ is the normal probability density and $\boldsymbol{\theta}$ is a vector of all parameters.

(a) Expain why Evil Robot has chosen this model for the data. [3 marks]

(b) Let $\mathbf{X}$ denote the set of m hits, which are assumed independent and identically distributed. Write down an expression for the log-likelihood of $\mathbf{X}$, conditional on the parameters. [2 marks]

(c) Denote the two components of $p(\mathbf{x}|\boldsymbol{\theta})$ as the 'wide' and 'narrow' components. Let the ith hit have an associated indicator $\mathbf{z}_i = (z_i^{\text{narrow}}, z_i^{\text{wide}})$ where

$$z_i^x = \begin{cases} 1 & \text{if } \mathbf{x}_i \text{ in component } x \\ 0 & \text{otherwise.} \end{cases}$$

Let $\mathbf{Z}$ denote the collection of the m random variables $\mathbf{z}_i$. Show that

$$\log p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) = \sum_{i=1}^{m} z_i^{\text{narrow}}[\log\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}, \sigma_1\mathbf{I}) + \log\pi] + z_i^{\text{wide}}[\log\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}, \sigma_2\mathbf{I}) + \log(1-\pi)].$$

[4 marks]

(d) The *EM algorithm* relies on the identity

$$\log p(\mathbf{X}|\boldsymbol{\theta}) = L(q, \boldsymbol{\theta}) + D_{\text{KL}}(q, p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}))$$

where q is an arbitrary distribution on $\mathbf{Z}$, D_{KL} is the Kullback-Leibler distance, and $L(q, \boldsymbol{\theta}) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \log \frac{p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta})}{q(\mathbf{Z})}$. Explain how this gives rise to the two steps of the EM algorithm. [3 marks]

(e) Derive the EM update for the parameter π. [8 marks]

9 Optimising Compilers

The following program in a C-like language is given to a compiler:

```
x = 42
a = x * y
b = a + z
c = b * 2

if (a > 0) {
  int d
  d = d - d
  e = d
}
print (e)
```

(*a*) (1-2) Name the two kinds of data-flow anomaly appearing in the code above. (3) Describe the data-flow analysis that can be used to identify them and (4) its corresponding data-flow equation(s). (5) State if this data-flow analysis is a forward or backward analysis and (6) explain how one can derive the direction (forward/backward) from the data-flow equation. [5 marks]

(*b*) Perform the data-flow analysis from part (*a*) on the code above and mark all data-flow anomalies that can be detected purely from the data-flow analysis without transforming the code. Include all details, e.g., list which properties you had to check to identify these data-flow anomalies. [5 marks]

(*c*) Draw the clash-graph of the code above (without removing any anomalies). Does the graph have specific properties that allow us to give an immediate upper bound for the number of colors needed to register-allocate the graph? Draw the smallest graph that requires 6 colors. [4 marks]

(*d*) In general, what is the computational complexity of graph colouring – of finding a colouring that uses the smallest number of colours? [2 marks]

(*e*) Describe a heuristic-based register-allocation approach and use it to allocate registers in the above program. Split ties by allocating variables earlier in the alphabet first. Variables assigned but unused should be allocated to a register named ‘zero’. Otherwise, use registers ‘r0’, ‘r1’, ‘r2’, and ‘r3’. [4 marks]

10 Principles of Communications

(*a*) Dynamic Alternative Routing is the official name for sticky random routing. Explain how this scheme decreases the chances of calls being blocked in the core of the old digital telephone network, while meeting the goal of simplicity. What assumptions are there about telephony traffic that mean the scheme works? In your answer, include the reason for trunk reservation. [10 marks]

(*b*) There are broadly two different classes of network traffic in terms of demands for performance guarantees, namely elastic and inelastic. Packet switched networks offer mechanisms to provide fairness and protection of flows from each other for both kinds of traffic.

Describe the mechanisms behind end-to-end congestion control and queue management that address fairness for elastic (closed loop controlled) traffic, and contrast with the approach of admission control and scheduling used for inelastic (real-time, open loop controlled) traffic flows. [10 marks]

11 Quantum Computing

Quantum gate teleportation. Consider a system with two data qubits (D_1, D_2) and two Bell state qubits (B_1, B_2). D_1 and D_2 are initialized to $a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ with a, b, c, d being suitably normalized. B_1 and B_2 are initialized to $(1/\sqrt{2})(|01\rangle + |10\rangle)$.

(a) Give a sequence of gates which can transform a state $|00\rangle$ to $(1/\sqrt{2})(|01\rangle + |10\rangle)$. [4 marks]

(b) Given a quantum state $\alpha|0\rangle + \beta|1\rangle$ where α and β are suitably normalized, let's measure it in the X basis (i.e., in the $|+\rangle$, $|-\rangle$ basis). What is the probability of measuring the state $|+\rangle$? What is the probability of measuring the state $|-\rangle$? [2 marks]

(c) The following operations are executed in sequence.

1. CNOT D_1, B_1 // Controlled NOT with control D_1 and target B_1
2. CNOT B_2, D_2
3. x = Measure B_1 in Z basis
4. y = Measure B_2 in X basis
5. If x is 0, apply X gate on D_2. Else apply I gate on D_2.
6. If y is 0 (i.e., the state is $|+\rangle$, apply I gate on D_1. Else apply Z gate on D_1.

Let's analyze how the state of the 4-qubit system changes as we execute the operations above. What is the state of the system after steps 1 and 2? For all states, use the ordering convention $|D_1 D_2 B_1 B_2\rangle$. [4 marks]

(d) Given particular values for x and y, what is the state of the system after steps 3 and 4? [4 marks]

(e) Prove that the overall effect of the sequence 1-6 is to apply a CNOT gate with D_1 as control and D_2 as target, up to global phase. [4 marks]

(f) Suppose B_1 and B_2 were initialized to the state $(1/\sqrt{2})(|00\rangle + |11\rangle)$, how should steps 5 and 6 be modified to realize the CNOT gate between D_1 and D_2? [2 marks]

12 Randomised Algorithms

Consider the following algorithm for finding the k-th smallest element of an array $A[1...n]$ consisting of n different natural numbers.

```
QUICK-SELECT(Array A[1...n], position k in [1..n])
   1 If n==1 return A[1]
   2 Pick an index 1 <= j <= n uniformly at random
   3 Let B[1...i] contain the numbers in A less than A[j]
   4 Let B[i+1...n-1] contain the numbers in A greater than A[j]
   5 If i+1==k then return A[j]
   6 else if i>=k then return QUICK-SELECT(B[1...i],k)
   7 else return QUICK-SELECT(B[i+1...n-1],k-i-1)
```

(*a*) For an input array $A[1 \dots n]$ of length n, what is the *maximum* (and *minimum*, respectively) number of recursive calls? [4 marks]

Now define the random variable $X_t \in \{0, 1, \dots, n\}$ to be the length of array A after t recursive calls; so $X_0 := n$. Further, if the algorithm does not make more than t recursive calls, we set $X_t := 0$.

(*b*) Assuming $n \geq 3$ is odd and $k = \lceil n/2 \rceil$, compute $\mathbb{E}[X_1]$. [4 marks]

(*c*) Let $n \geq 2$ and $1 \leq k \leq n$ be arbitrary. Assume that there exists $\epsilon > 0$ such that for any $m \geq 1$ and any $t \geq 1$, the inequality $\mathbb{E}[X_t \mid X_{t-1} = m] \leq (1 - \epsilon) \cdot m$ holds.

(*i*) Which bound can you deduce for any $\mathbb{E}[X_t]$, where $t \geq 1$? [4 marks]

(*ii*) Prove an upper bound on the maximum number of recursive calls that holds with probability at least $1 - n^{-1}$. [4 marks]

In the following, we will assume that the computation of the two arrays B in lines 3-4 can be performed in $O(n)$ time.

(*d*) Using earlier parts, how could you express the time complexity of the algorithm? State an upper bound on the expected time complexity. [4 marks]

13 Types

(*a*) For each of the types below, give a well-typed term inhabiting it in the simply-typed lambda calculus augmented with sums, products and the letcont/throw primitives for continuations.

(*i*) $\mathsf{lem}_A : A + \neg A$ [3 marks]

(*ii*) $\mathsf{abs} : \neg(A \times B) \to A \to \neg B$ [5 marks]

(*iii*) $\mathsf{dem} : \neg(A \times B) \to (\neg A) + (\neg B)$ [5 marks]

(*b*) Consider the following function in the simply-typed lambda calculus augmented with lists, booleans, and the letcont/throw primitives for continuations.

$$
\begin{array}{lcl}
\mathsf{all} & : & (X \to \mathsf{bool}) \to \mathsf{List}\, X \to \mathsf{bool} \\
\mathsf{all}\, p\, xs & = & \lambda p : X \to \mathsf{bool}.\, \lambda xs : \mathsf{List}\, X. \\
 & & \quad \mathsf{letcont}\ k : \neg\mathsf{bool}\ \mathsf{in} \\
 & & \quad \mathsf{fold}(xs, [] \to \mathsf{true}, y :: r \to \mathsf{if}\ p(x)\ \mathsf{then}\ r\ \mathsf{else}\ \mathsf{throw}(k, \mathsf{false}))
\end{array}
$$

In English, explain the operational behaviour of this function. [4 marks]

(*c*) (*i*) Suppose we encode $n \times m$ matrices in Agda with the type $\mathsf{Vec}\ (\mathsf{Vec}\ \mathbb{R}\ m)\ n$. Give an Agda type for a function that performs matrix transposition. [1 mark]

(*ii*) Give an Agda type expressing that addition on the natural numbers is commutative. [2 marks]

END OF PAPER