CST1
COMPUTER SCIENCE TRIPOS Part IB

Wednesday 10 June 2026 14:00 to 17:00

COMPUTER SCIENCE Paper 6

*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 Complexity Theory

Let $A \leq_p B$ denote polynomial-time many-one reducibility.

(*a*) Define $A \leq_p B$. What does it mean for a language to be NP-complete? [2 marks]

(*b*) Define K-SAT and show that 2SAT$\in$ P. [4 marks]

(*c*) VERTEX-COVER is defined as follows: given a graph $G = (V, E)$ and integer k, decide whether there exists $S \subseteq V$ with $|S| \leq k$ such that every edge has at least one endpoint in S. Prove that VERTEX-COVER$\in$ NP. [2 marks]

(*d*) Give a polynomial-time reduction from 3SAT to VERTEX-COVER, and prove its correctness. [8 marks]

(*e*) Define INDEPENDENT-SET and prove that it is NP-complete by giving a polynomial-time reduction from VERTEX-COVER. [4 marks]

2 Complexity Theory

Let GI be the language

$$\mathsf{GI} = \{(G_1, G_2) : G_1 \cong G_2\},$$

where G_1, G_2 are simple graphs on the same number of vertices.

Consider the following protocol for GI. On common input (G_1, G_2), the prover P (who knows an isomorphism $\varphi : G_1 \to G_2$ if one exists) does the following:

- P chooses a uniformly random permutation π of the vertices and sends $H = \pi(G_1)$.
- V chooses a uniformly random bit $b \in \{1, 2\}$ and sends b.
- P sends an isomorphism $\sigma : G_b \to H$ (if $b = 1$, $\sigma = \pi$; if $b = 2$, $\sigma = \pi \circ \varphi^{-1}$). V accepts iff $\sigma(G_b) = H$.

Answer the following questions, providing complete definitions and proofs.

(a) Define the notion of *statistical zero knowledge proofs*, and explain how the notion of a simulator helps capture the zero knowledge property. [5 marks]

(b) Define *honest-verifier* zero knowledge (HVZK) and explain briefly why it is a weaker notion than full zero knowledge. [3 marks]

(c) Prove completeness and soundness of the protocol for GI stated above. [6 marks]

(d) Show that the protocol is HVZK by describing a simulator, and then show that it produces the same distribution as the honest verifier's view. [6 marks]

3 Computation Theory

A *Gödel numbering* of register machines is a bijection G between the natural numbers $\mathbb{N}$ and the collection of register machines.

(a) Give an example of a Gödel numbering of register machines. [5 marks]

The *Halting Problem* for register machines (with respect to a Gödel numbering G) is a set of pairs of numbers $H \subseteq \mathbb{N} \times \mathbb{N}$.

(b) Give a precise statement of what pairs of numbers constitute H, given the Gödel numbering from Part (a). [3 marks]

(c) Give a precise statement of what it means to say that H is *undecidable*. [2 marks]

Consider the following set of numbers

$$Z = \{n \in \mathbb{N} \mid G(n) \text{ halts when started with 0 in all registers}\}$$

(d) Describe a computable function $r : \mathbb{N}^2 \to \mathbb{N}$ such that for all $m, n \in \mathbb{N}$ we have $(m, n) \in H$ if, and only if, $r(m, n) \in Z$ and show that it has these properties. [7 marks]

(e) What can you conclude about the decidability or otherwise of Z? Give justification for your answer. [3 marks]

4 Computation Theory

(*a*) What is the class of *primitive recursive functions*? [2 marks]

(*b*) Show that the following function $\mathrm{nz} : \mathbb{N} \to \mathbb{N}$ is primitive recursive:

$$\mathrm{nz}(x) \triangleq \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{otherwise.} \end{cases}$$

[3 marks]

For any function $f : \mathbb{N}^n \to \mathbb{N}$, let $\mathrm{nz}_f : \mathbb{N}^n \to \mathbb{N}$ be the function:

$$\mathrm{nz}_f(\bar{x}) \triangleq \begin{cases} 0 & \text{if } f(\bar{x}) = 0 \\ 1 & \text{otherwise.} \end{cases}$$

(*c*) Show that if f is primitive recursive, then so is nz_f. [2 marks]

For any function $g : \mathbb{N}^{n+1} \to \mathbb{N}$, let $\mathrm{prod}_g : \mathbb{N}^{n+1} \to \mathbb{N}$ be the function:

$$\mathrm{prod}_g(\bar{x}, y) \triangleq \prod_{i=0}^{y} g(\bar{x}, i).$$

(*d*) Show that if g is primitive recursive, then so is prod_g. [5 marks]

For any function $h : \mathbb{N}^{n+1} \to \mathbb{N}$, let $\mathrm{bex}_h : \mathbb{N}^{n+1} \to \mathbb{N}$ be the function where $\mathrm{bex}_h(\bar{x}, y) = 0$ if there is no $i \leq y$ with $h(\bar{x}, i) = 0$ and $\mathrm{bex}_h(\bar{x}, y) = 1$ if there is such an i.

(*e*) Show that if h is primitive recursive, then so is bex_h. [4 marks]

(*f*) Now let $\mathrm{ex}_h : \mathbb{N}^n \rightharpoonup \mathbb{N}$ be the partial function where $\mathrm{ex}_h(\bar{x}) = 1$ if there is an x such that $h(\bar{x}, x) = 0$ and $\mathrm{ex}_h(\bar{x})$ is undefined otherwise. Give an example of a primitive recursive h for which ex_h is not primitive recursive. [4 marks]

5 Data Science

The m members of the Cambridge University Quiddlefinks Society have been playing matches against each other all term. It's your job as secretary to use match data to estimate the skill level s_a of each player $a \in \{1, \ldots, m\}$.

A quiddlefinks match is between two players, and has a single winner. For match $i \in \{1, \ldots, n\}$ we record the two players $a_i, b_i \in \{1, \ldots, m\}$, and also the winner $y_i \in \{0, 1\}$, where $y_i = 1$ means that player a_i won and $y_i = 0$ means that player b_i won.

(*a*) Consider a match between players $a, b \in \{1, \ldots, m\}$. Suggest a probability model for $Y = 1_{a \text{ wins}}$ given skill levels s_a and s_b. Write down the log likelihood of the full list of match outcomes $y_1, \ldots, y_n$. [5 marks]

(*b*) Explain how to estimate the parameters $s_1, \ldots, s_m$. Give pseudocode. [3 marks]

(*c*) Alternatively we can take a Bayesian approach and let the skill levels S_a, $a \in \{1, \ldots, m\}$, be random variables. Take as prior that skills are independent $N(0, 1)$. Explain how to plot the posterior distribution of the skill S_a of a given player a. [5 marks]

(*d*) You are asked to predict the outcome of an upcoming match between players a^* and b^*. Give an expression for the probability that a^* will win, in terms of the skill levels S_{a^*} and S_{b^*}, according to your model from Part (*a*). [2 marks]

(*e*) Explain how to find a 95% posterior confidence interval for the probability you found in Part (*d*). [5 marks]

6 Data Science

A social scientist is studying social mobility. For each individual i in the dataset, she has a record of their education level e_i, their income q_i, and their parents' income p_i. For simplicity she takes education level to be a binary variable, $e_i \in \{0, 1\}$. Treat these observations as independent samples of random variables (E, Q, P).

(a) Propose a model for E as a function of P, and explain how to fit it. Your model must treat E as binary, not as real-valued. [6 marks]

(b) Propose a model for Q as a function of E, and explain how to fit it. [4 marks]

(c) Propose a model for Q as a function of both P and E, and explain how to fit it. [4 marks]

(d) The social scientist believes that in an ideal world one's income should depend only on one's educational level. She wishes to test the hypothesis that this dataset is consistent with her ideal world. Explain how to test this hypothesis. Give pseudocode. [6 marks]

7 Logic and Proof

(a) Consider the following statement:

$$\forall x \forall y \forall z (R(x,y) \wedge R(y,z) \rightarrow R(x,z)) \wedge \forall x \exists y R(x,y) \rightarrow \forall x R(x,x).$$

Prove that it is valid using resolution or find a counterexample. Show all steps: negation, skolemisation, conversion to clauses, and the resolution proof. If a counterexample exists, explain how resolution is blocked from producing a spurious proof.

[9 marks]

(b) Use the DPLL method to find a model satisfying the following set of clauses, or to prove that no such model exists. Show each step of the method clearly.

$$\{P, Q, \neg R\} \quad \{\neg P, R\} \quad \{R, \neg Q, \neg R\} \quad \{\neg Q, R\} \quad \{P, \neg Q\} \quad \{\neg P, \neg R\}$$

[6 marks]

(c) Apply the Fourier-Motzkin variable elimination method to determine the satisfiability of the following system of linear inequalities over real numbers.

(i) $x + y + z \leq 1$

(ii) $x + y - z \leq -1$

(iii) $x \geq 2$

If the system is satisfiable, give an assignment to the variables that satisfies all inequalities (a model). If it is not satisfiable, derive a contradiction.

[5 marks]

8 Logic and Proof

(*a*) Consider an automated traffic management system for a narrow bridge with one-way traffic only. Cars may enter the bridge only from one side, which is controlled by a traffic light. Let G mean the light is green, R mean the light is red, and M mean a car is moving on the bridge. When the light is red, cars do not enter the bridge. Formalise the following requirements in S4 modal logic:

(*i*) It is necessarily true that if the light is green, it is possible for a car to be moving.

(*ii*) It is necessarily true that if a car is moving, then it is not possible for the light to be red.

(*iii*) Henceforth, eventually the light becomes green.

[3 marks]

(*b*) Using your formalisation from Part (*a*), show that the following formula holds:

$$\Box(M \rightarrow \neg R).$$

[3 marks]

(*c*) Provide a formal proof in the S4 sequent calculus for the following sequent:

$$\Box(P \rightarrow \Diamond Q) \Rightarrow \Diamond P \rightarrow \Diamond\Diamond Q$$

[7 marks]

(*d*) In S4 modal logic, the accessibility relation is reflexive and transitive. Define a new operator $\blacksquare$ such that $\blacksquare A \equiv A \wedge \Box A$.

(*i*) Derive the left and right sequent rules for $\blacksquare$ based on the rules for $\wedge$ and $\Box$.

(*ii*) In the context of S4, explain whether $\blacksquare A \simeq \Box A$ holds.

[7 marks]

9 Semantics of Programming Languages

Some programmers believe that C pointers behave just as machine-word addresses. That's not true for actual C, but it's interesting to explore what some of the consequences would be if it were.

(a) Give a semantics with machine-word (64-bit integer) pointers for the following vaguely C-ish language in which function arguments are mutable, allocated to fresh machine-word addresses on function entry, $\&x$ denotes the address allocated for x, and dereferencing and update are allowed on any memory location. The semantics should define a small-step transition relation $A, M, e \longrightarrow A', M', e'$, where memory $M : \mathbb{N}_{64} \to \mathbb{N}_{64}$ holds a 64-bit integer at each 64-bit integer address, and an allocation environment $A \subseteq \mathbb{N}_{64}$ records which addresses have been allocated so far.

$$e ::= \quad n \mid e + e' \mid \&x \mid *e \mid *e = e' \mid e; e' \mid \mathbf{fun}\,(x)\{e\} \mid e\; e'$$

In $\mathbf{fun}\,(x)\{e\}$ the x binds in e. [9 marks]

(b) Describe all the configurations in which your semantics gets stuck. [2 marks]

(c) Give two examples in which a function can observably modify the function argument of its caller, one syntactically obviously and one indirectly, and explain them briefly. [2 marks]

(d) Comment on situations in which optimisations that move code, such as common subexpression elimination, are invalid in your semantics, with an example. [1 mark]

(e) (i) Define a static type system for this language that prevents confusion of integer and pointer values, where the types of machine words are

$$T_{\mathrm{w}} ::= \quad \mathbf{int} \mid T_{\mathrm{w}}*$$

the types of expressions are

$$T ::= \quad T_{\mathrm{w}} \mid T_{\mathrm{w}} \to T'_{\mathrm{w}}$$

and functions are type-annotated $\mathbf{fun}\,(x : T_{\mathrm{w}})\{e : T'_{\mathrm{w}}\}$. [4 marks]

(ii) State which of your Part (b) configurations are ruled out by the type system. [2 marks]

END OF PAPER