CST1
COMPUTER SCIENCE TRIPOS Part IB

Wednesday 11 June 2025 13:30 to 16:30

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 SSP be the Subset Sum Problem, defined as follows.
Instance: A finite set of integers S and a target integer t.
Problem: Does there exist a subset $S' \subseteq S$ such that $\sum_{s \in S'} s = t$?

(a) Provide two definitions of the class NP: one in terms of non-determinism and the other in terms of verification. Prove that the two notions are equivalent. [6 marks]

(b) Show that SSP $\in$ NP. [2 marks]

(c) Prove that SSP is NP-complete. (*Hint: to establish hardness, you can reduce from the NP-complete 3D Matching Problem, defined as follows. Given disjoint sets X, Y, Z, each of size n, and a collection of triples $T \subseteq X \times Y \times Z$, determine whether there exists a subset $M \subseteq T$ of size n such that each element of X, Y, and Z appears in exactly one triple of M.*) [8 marks]

(d) Suppose that one finds a non-deterministic logspace algorithm for SSP. What would this imply about the NP vs co-NP problem? [4 marks]

2 Complexity Theory

(a) Define the class BPP formally. [2 marks]

(b) Show that the standard choice of error probability bound in the definition of BPP can be reduced to an exponentially small error. [4 marks]

(c) Suppose we set the error probability bound to only require less than 1/2 (i.e., success probability at least 1/2) in the above definition. Would that change the resulting complexity class? [6 marks]

(d) Prove that BPP $\subseteq$ P/Poly. [4 marks]

(e) Is BPP = P/Poly? Justify your answer. [4 marks]

3 Computation Theory

There are many ways to represents lists in the λ-calculus. You will explore one of them here. Let $\underline{n}$ be a λ-term representing the integer n. For this question, integer lists are represented as

$$\begin{aligned} \underline{[]} &= \lambda x\ f.\ x \\ \underline{[n_1,\ n_2,\ \ldots,\ n_m]} &= \lambda x\ f.\ f\ \underline{n_1}(f\ \underline{n_2}\ \ldots\ (f\ \underline{n_m}\ x)\ \ldots) \end{aligned}$$

(*a*) Present, with justification, a λ-term $\mathbf{H}$ such that we have

$$\mathbf{H}\ \underline{[n_1,\ n_2,\ \ldots, n_m]} =_\beta \underline{n_1}.$$

[2 marks]

(*b*) Present, with justification, a λ-term $\mathbf{L}$ such that for any list l we have

$$\underline{\text{length } l} =_\beta \mathbf{L}\ \underline{l}.$$

[4 marks]

(*c*) Suppose a function g is represented with the λ-term $\mathbf{G}$. Present, with justification, a λ-term $\mathbf{M}$ such that for any list l we have

$$\underline{\text{map } g\ l} =_\beta \mathbf{M}\ \mathbf{G}\ \underline{l}.$$

[4 marks]

(*d*) Consider this Ocaml code for reversing a list:

```
let rec aux r l =
    if l = []
    then r
    else aux ((hd l) :: r) (tl l)

let rev = aux []
```

In answering the following questions you may use these λ-terms and facts.

$$\begin{array}{rcl|rcl} \mathbf{Y}\ F & =_\beta & F\ (\mathbf{Y}\ F) & \mathbf{IF}\ \mathbf{TRUE}\ F\ S & =_\beta & F \\ \mathbf{N}\ \underline{[]} & =_\beta & \mathbf{TRUE} & \mathbf{IF}\ \mathbf{FALSE}\ F\ S & =_\beta & S \\ \mathbf{N}\ \underline{[n_1,\ \ldots,\ n_m]} & =_\beta & \mathbf{FALSE} & \mathbf{C}\ \underline{n}\ \underline{l} & =_\beta & \underline{n :: l} \end{array}$$

(*i*) Define λ-terms $\mathbf{A}$ representing `aux` and $\mathbf{R}$ representing `rev`. [4 marks]

(*ii*) Prove that $\mathbf{R}\ \underline{l}$ correctly implements list reversal for every list l.

[6 marks]

4 Computation Theory

These questions deal with the theory of partial recursive functions.

Given a subset $S \subseteq \mathbb{N}$, its characteristic function $\chi_S \in \mathbb{N} \to \mathbb{N}$ is given by

$$\chi_S(x) = \begin{cases} 1 & x \in S \\ 0 & z \notin S. \end{cases}$$

We say that S is *decidable* if χ_S is a total recursive function.

A set S will be called *recursively enumerable* if it is empty or there is a total recursive function f such that

$$S = \{f(n) \mid n \in \mathbb{N}\}.$$

Note that it may be that $f(n) = f(m)$ for some $m \neq n$. That is, the enumeration may involve repeats.

(*a*) Are all total partial recursive functions primitive recursive? [2 marks]

(*b*) Suppose that $S \subseteq \mathbb{N}$ is decidable. Prove that S is recursively enumerable. [6 marks]

(*c*) Prove that a set $S \subseteq \mathbb{N}$ is decidable if and only if both S and its complement $\overline{S}$ are recursively enumerable. [6 marks]

(*d*) Provide a complete proof that the set $S_0 = \{e \mid \phi_e(0)\downarrow\}$ is recursively enumerable but its compliment is not. [6 marks]

5 Data Science

(*a*) We have two groups of paired data, (a_j, b_j) for $j \in \{1, ..., m\}$ and (a'_k, b'_k) for $k \in \{1, ..., m'\}$. Consider the probability model

$$B_j \sim \text{Poisson}(\lambda a_j)\,, \qquad B'_k \sim \text{Poisson}(\mu a'_k)$$

where λ and μ are unknown parameters.

(*i*) Find maximum likelihood estimates for λ and μ. (*ii*) Describe how to test the hypothesis that $\lambda = \mu$. [5 marks]

(*b*) We have a dataset in which each datapoint $i \in \{1, ..., n\}$ consists of an integer response y_i and a collection of m non-negative features $(e_{1,i}, ..., e_{m,i})$. A *Poisson-linear* model is a model for this dataset of the form

$$Y_i \sim \text{Poisson}\Bigl(\sum_{k=1}^{m} \beta_k e_{k,i}\Bigr)$$

where $\beta_1, ..., \beta_m$ are unknown parameters, assumed to be strictly positive.

(*i*) Give pseudocode for fitting a Poisson-linear model. (*ii*) Express the model from part (*a*) as a Poisson-linear model. [4 marks]

(*c*) In a factory, the workers suspect that one particular manager is incompetent, since there tend to be more safety incidents on days when that manager is present. They have assembled a dataset spanning several months, giving for each day the number of safety incidents and the names of the managers present. They believe that each manager has a competence level, and the number of incidents on a given day is related to the mean competence level among the managers present.

(*i*) Suggest a probability model for this data, in the form of a Poisson-linear model. (*ii*) Explain how to test whether the suspect manager is indeed worse than the others. [11 marks]

6 Data Science

This question concerns a machine for making widgets. The machine is unreliable, and the proportion of defective widgets varies from day to day. For a given day, let Θ be the probability that a given widget is defective, and assume that widgets that day are conditionally independent given Θ.

You may give either analytical or computational answers. If you give computational answers, you must provide clearly-commented pseudocode.

(*a*) You sample n widgets, and find that x are defective. Find a posterior confidence interval for Θ, given the prior belief that $\Theta \sim \text{Beta}(\alpha, \beta)$. [4 marks]

(*b*) An alternative sampling strategy is to keep sampling widgets until you find the first defective widget. Let x be the number of widgets you end up sampling. Find a posterior confidence interval for Θ, given the prior belief that $\Theta \sim \text{Beta}(\alpha, \beta)$. [3 marks]

(*c*) For the sampling strategy in part (*b*), find the expected number of samples needed. Assume that α and β are positive integers with $\alpha > 1$; your answer should be a function of α and β. [4 marks]

(*d*) An engineer tells you that on a given day, the machine is either healthy or cranky, each equally likely, and that $\Theta \sim \text{Beta}(\alpha_m, \beta_m)$ where $m \in \{h, c\}$ indicates whether the machine is healthy or cranky. Find the posterior probability that the machine is cranky, for each of the two sampling strategies above. [9 marks]

[*Note:* The Beta distribution has density $\Pr(x; \alpha, \beta) = x^{\alpha-1}(1-x)^{\beta-1}/B(\alpha, \beta)$ where, for integer α and β, $B(\alpha, \beta) = (\alpha-1)!(\beta-1)!/(\alpha+\beta-1)!$]

7 Logic and Proof

(*a*) Present either a proof in tableaux or a falsifying interpretation for:

$$\forall x(P(x) \to Q(x)) \to (\forall y P(y) \to \forall z Q(z))$$

[11 marks]

(*b*) Convert the following formulae (where a and b are constants, and x, y and z are variables) into clauses, and exhibit a model or show that none exists using resolution.

$$\begin{aligned}
P(x,x) &\;\to\; R(a,x) && \text{(i)}\\
P(x,y) \wedge P(y,x) &\;\to\; Q(x) && \text{(ii)}\\
Q(y) &\;\to\; \neg Q(a) && \text{(iii)}\\
R(b,y) &\;\to\; P(y,y) && \text{(iv)}\\
\neg R(x,y) &\;\to\; R(z,a) && \text{(v)}
\end{aligned}$$

[9 marks]

8 Logic and Proof

(*a*) Present either a proof in sequent calculus or a falsifying interpretation for:

$$\forall x(P(x) \to Q(x)) \to (\exists y(P(y) \wedge R(y)) \to \exists z(Q(z) \wedge R(z)))$$

[11 marks]

(*b*) Draw alphabetically ordered Binary Decision Diagrams (BDDs) for:

$$F_1 = (P \wedge Q) \vee \neg T \text{ and } F_2 = (\neg R \vee \neg S) \wedge (R \vee \neg T).$$

Then draw a BDD for $F_1 \vee F_2$.

[9 marks]

9 Semantics of Programming Languages

This mini-C language has mutable variables and block-structured scope:

expression, $e ::= \;\; n \mid id \mid id = e \mid e\,;e' \mid \{\,\texttt{int}\; id_1\,;\, ..\, \texttt{int}\; id_i\,;\, e\}$

Its operational semantics can be expressed in terms of environments E that are partial functions mapping identifiers id to the numeric addresses $n \in \mathbb{N}$ they are allocated at, memory heaps H that are partial functions from addresses $\mathbb{N}$ to values $\mathbb{N}$, atomic evaluation contexts $A ::= \;\; id = \text{-} \mid \text{-}\,; e'$, evaluation contexts $C ::= \;\; \text{-} \mid C \cdot A$ which are lists of those, stacks $S ::= \;\; \texttt{nil} \mid F :: S$ which are lists of a stack frame for each enclosing block, where a stack frame $F ::= \;\; \langle C, E\rangle$ consists of an evaluation context and the environment for that block's local variables, and configurations $\langle e,\, S,\, H\rangle$. When we combine partial functions, e.g. with H, H', their domains must be disjoint. Initial configurations are $\langle e,\, \langle \text{-}, \mathrm{emp}\rangle :: \texttt{nil},\, \mathrm{emp}\rangle$, writing emp for empty partial functions.

$$\frac{\text{lookup } S\; id = n \qquad H(n) = n'}{\langle id, S, H\rangle \rightarrow \langle n', S, H\rangle}\ \text{VAR} \qquad \frac{\text{lookup } S\; id = n}{\langle id = n', S, (H, n \mapsto n_0)\rangle \rightarrow \langle n', S, (H, n \mapsto n')\rangle}\ \text{ASSIGN}$$

$$\frac{}{\langle n\,; e, S, H\rangle \rightarrow \langle e, S, H\rangle}\ \text{SEQ\_INT}$$

$$\frac{E' = id_1 \mapsto n_1,\, ..,\, id_i \mapsto n_i \qquad H' = n_1 \mapsto 0,\, ..,\, n_i \mapsto 0}{\langle \{\,\texttt{int}\; id_1\,;\, ..\, \texttt{int}\; id_i\,;\, e\}\;\; , S, H\rangle \rightarrow \langle e, \langle \text{-}, E'\rangle :: S, (H, H')\rangle}\ \text{BLOCK\_START}$$

$$\frac{}{\langle n, (\langle \text{-}, E\rangle :: S), H\rangle \rightarrow \langle n, S, H \setminus \mathrm{ran}\,(E)\rangle}\ \text{BLOCK\_END}$$

$$\frac{\neg\,(\mathrm{isvalue}\,(e))}{\langle A[e], (\langle C, E\rangle :: S), H\rangle \rightarrow \langle e, (\langle C \cdot A, E\rangle :: S), H\rangle}\ \text{EVAL\_CTX\_FOCUS}$$

$$\frac{}{\langle n, (\langle C \cdot A, E\rangle :: S), H\rangle \rightarrow \langle A[n], (\langle C, E\rangle :: S), H\rangle}\ \text{EVAL\_CTX\_DEFOCUS}$$

(*a*) Define the lookup function. [4 marks]

(*b*) Give the transition sequence, with the configuration and rule name for each transition, of $\langle \{\,\texttt{int}\,\texttt{x}\,;\texttt{x} = \texttt{1}\}\;\; ;\texttt{y},\, \langle \text{-}, \mathrm{emp}\rangle :: \texttt{nil},\, \mathrm{emp}\rangle$. Include a brief explanation alongside each transition. [10 marks]

(*c*) Explain, with examples and reference to the rules, but without giving transition sequences, how this language treats variable shadowing. [4 marks]

(*d*) Explain what changes would be needed to add global variables. [2 marks]

END OF PAPER