

COMPUTER SCIENCE TRIPOS Part IB

Tuesday 2 June 2009 1.30 to 4.30

COMPUTER SCIENCE Paper 4

*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.*

<p>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</p>

STATIONERY REQUIREMENTS

*Script paper**Blue cover sheets**Tags*

SPECIAL REQUIREMENTS

Approved calculator permitted

1 Algorithms II

- (a) Explain the terms *amortized analysis* and *aggregate analysis*, highlighting the difference between them. [3 marks]
- (b) Assume that the following two classes, which implement the standard stack (last-in first-out) and queue (first-in first-out) data structures, are available.

```

class Stack
    void push(Item x)
    Item pop()
    Boolean isEmpty()

class Queue
    void enqueue(Item x)
    Item dequeue()
    Boolean isEmpty()

```

- (i) A `Multistack` class is derived from `Stack` with the addition of two methods:
- a `void multipush(Itemlist l)` that takes a list `l` of items and pushes each of the items onto the stack (each action of extracting an item from the list and pushing it onto the stack has constant cost), and
 - a `void multipop(int m)` that takes an integer `m` and pops that many items off the stack (raising an exception if there were fewer, but don't worry about that).

Is it true or false that, given an arbitrary sequence of n `Multistack` operations starting from an empty `Multistack`, each operation in the sequence has amortized constant cost? Justify your answer in detail.

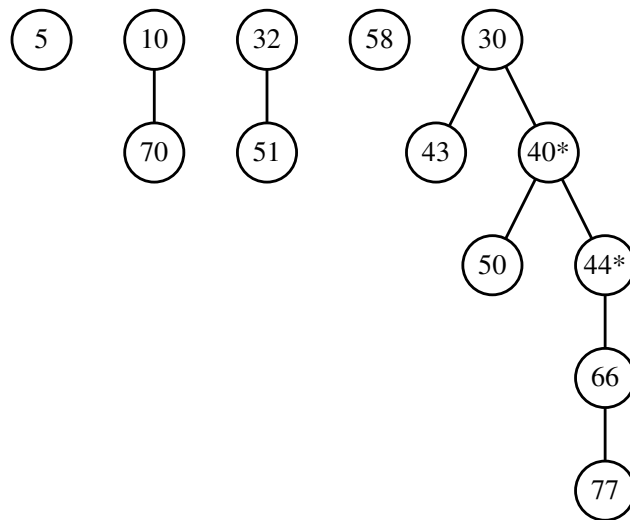
[5 marks]

- (ii) Provide an implementation of class `Queue` using no other data structures than `Item`, `Boolean`, `int` and `Stack`. The amortized running time of each `Queue` method must be constant. (*Note that you may only use the `Stack` as a black box: you are not allowed to access its internal implementation.*) [7 marks]
- (iii) Using the potential method, prove that the amortized running time of all your `Queue` methods from part (b)(ii) is indeed constant. [5 marks]

2 Algorithms II

- (a) Given the following Fibonacci heap, where nodes with an asterisk are “marked”, perform `extractMin()` on it and then `decreaseKey()` on the node whose key is currently 66, bringing it down to 4. Redraw the changed heap as you go along.

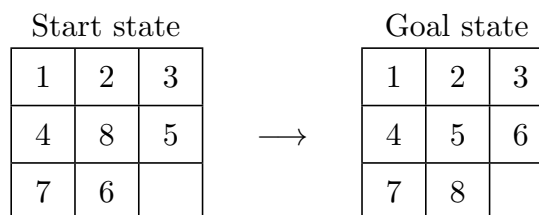
You need only draw any significant intermediate states of the heap, adding any necessary explanations so that a reader can follow what you are doing and why. [5 marks]



- (b) Fibonacci heaps are designed so that their trees never become excessively “wide and shallow”. Why? Justify this design goal in detail and then explain how it is achieved. [5 marks]
- (c) Nothing, however, stops the trees in a Fibonacci heap from growing “tall and narrow”. Prove this by describing a sequence of Fibonacci heap operations that, given an integer n , produces a Fibonacci heap made of a single tree consisting of a linear chain of n nodes (in other words, each node in the tree except for the last one is the parent of exactly one node, and each node except for the first one is the child of exactly one node). [10 marks]

3 Artificial Intelligence I

- (a) Explain the difference between *uninformed* and *informed* search. List *two* examples of each type of algorithm. [2 marks]
- (b) In the context of planning, describe what a *heuristic* is and what it means for it to be *admissible*. List *two* examples of typical heuristic functions. [Hint: consider the problem in part (d) below.] [2 marks]
- (c) Explain what A* search is, including the advantages and disadvantages with respect to its theoretical properties. [3 marks]
- (d) Draw a search tree for the 8-puzzle problem up to depth 4 (start state is depth 0) using the A* algorithm (omit repeated states) with the evaluation function $f(n) = p(n) + h(n)$, where $p(n)$ is the number of steps from the start state (start state is step 0) and $h(n)$ is the number of misplaced tiles. Note that the actions for sliding tiles should be used in this order: right, left, up and down. Write the values of f and of its components p and h under each state. You may use an abbreviated notation indicating only the tiles that change. [10 marks]



- (e) Briefly explain IDA* search and its advantages and disadvantages. What happens when using IDA* in the search problem in part (d) if the IDA* limit is 3? What happens if the limit is 4 (in terms of number of states)? [3 marks]

4 Artificial Intelligence I

- (a) Describe the following in the context of planning:
- (i) STRIPS operators; [1 mark]
 - (ii) situation space *versus* plan space (point out the differences); [2 marks]
 - (iii) partial order planning (briefly explain key features). [2 marks]
- (b) Consider the following partial order planning problem for creating a picture of an aquarium. The goal is to have `painted_background` and also `drawn(Fish)`, `drawn(Crab)` and `drawn(Seahorse)`. The start state is `empty(Picture)`. You can use the following:
- `paint_background` with precondition `empty(Picture)` and with effects `painted_background` and \neg `empty(Picture)`
 - `draw(x)` with no preconditions and with effects `drawn(x)` and \neg `empty(Picture)`
- (i) Write the *start* state and the *finish* state. Draw a partial order plan with preconditions above the operators and effects below the operators. Draw the causal links. [8 marks]
 - (ii) Define what threats are. Comment on whether there are any in your partial order plan above, and how you would solve them. Add temporal links to your partial plan as dotted arcs. [4 marks]
 - (iii) How many solution plans are there in your partial order plan? List all solution plans with total ordering of steps implied by your partial order plan. [3 marks]

5 Computer Graphics and Image Processing

- (a) Describe an algorithm that draws a Bézier cubic curve to a specified tolerance using straight lines. [7 marks]
- (b) Describe an algorithm for clipping a line against a rectangle. [7 marks]
- (c) A Bézier cubic curve could be clipped and drawn using the algorithm in part (a) to produce straight lines and the algorithm in part (b) to do the clipping. Describe a more efficient algorithm that draws a Bézier cubic curve clipped against a rectangle. [6 marks]

6 Computer Graphics and Image Processing

- (a) Describe the A-buffer polygon scan conversion algorithm using 4×4 sub-pixels for each pixel. [10 marks]
- (b) It is possible to represent continuous tone greyscale images using just black ink on white paper because of limitations in the human visual system. Explain how and why. [4 marks]
- (c) Describe an algorithm that, given a greyscale image, will produce a black and white (bi-level) image of four times the resolution in each dimension which provides a good approximation to the greyscale image. [6 marks]

7 Databases

- (a) Define Boyce–Codd Normal Form (BCNF). [3 marks]
- (b) Suppose that $R(A, B)$ is a relational schema with two attributes, A and B . Show that this schema must be in BCNF. [6 marks]
- (c) Describe a procedure that can take as input any relation schema $R(X)$ together with functional dependencies F over the attributes of the set X , and return a decomposition of the input schema where each relation is in BCNF. Make sure you argue that the procedure is correct and terminates. [8 marks]
- (d) Does your BCNF decomposition procedure always preserve all functional dependencies? Explain your answer. [3 marks]

8 Databases

(a) Define the concept of a *functional dependency*. [2 marks]

(b) Let $R(A, B, C, D, E, F)$ be a database schema with functional dependencies

$$\begin{aligned} A, B &\rightarrow C \\ B, C &\rightarrow A, D \\ D &\rightarrow E \\ C, F &\rightarrow B \end{aligned}$$

(i) Compute the closure of $\{A, B\}$. [3 marks]

(ii) Is $A, B \rightarrow D, F$ a functional dependency over R ? Justify your answer. [1 mark]

(c) Define the concept of a *multivalued dependency*. [2 marks]

(d) Suppose the functional dependency $X \rightarrow Y$ holds on a relational schema. Does this mean that the multivalued dependency $X \twoheadrightarrow Y$ holds? Justify your answer. [3 marks]

(e) Define the concept of a *lossless-join decomposition*. [3 marks]

(f) Let $R(X)$ be a database schema, where X is a set of attributes. Show that $S(Y \cup Z)$ and $T(Y \cup (X - Z))$ is a lossless-join decomposition of $R(X)$ if and only if the multivalued dependency $Y \twoheadrightarrow Z$ holds over R . [6 marks]

9 Mathematical Methods for Computer Science

For functions $f : \mathbb{R} \rightarrow \mathbb{C}$ define the *Fourier transform* of f , written $\mathcal{F}_{[f]}(\omega)$, as the function $\mathbb{R} \rightarrow \mathbb{C}$ given by

$$\mathcal{F}_{[f]}(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx$$

whenever the integral exists.

(a) Show that

$$\mathcal{F}_{[\alpha f + \beta g]}(\omega) = \alpha \mathcal{F}_{[f]}(\omega) + \beta \mathcal{F}_{[g]}(\omega)$$

for all complex constants $\alpha, \beta \in \mathbb{C}$ and functions f and g such that their Fourier transforms exist. [4 marks]

(b) For $a, b \in \mathbb{R}$ with $a \neq 0$, let $g(x) = f(ax + b)$. Show that

$$\mathcal{F}_{[g]}(\omega) = \frac{1}{|a|} e^{i\omega b/a} \mathcal{F}_{[f]} \left(\frac{\omega}{a} \right)$$

[6 marks]

(c) Derive $\mathcal{F}_{[f_c]}(\omega)$ where

$$f_c(x) = \begin{cases} 1 & -c < x \leq c \\ 0 & \text{otherwise} \end{cases}$$

and c is a positive constant.

[4 marks]

(d) For the piecewise constant function

$$h(x) = \begin{cases} 0 & x > 4 \\ 2 & 3 < x \leq 4 \\ 1 & 2 < x \leq 3 \\ 3 & 1 < x \leq 2 \\ 0 & x \leq 1 \end{cases}$$

derive its Fourier transform $\mathcal{F}_{[h]}(\omega)$.

[6 marks]

10 Mathematical Methods for Computer Science

- (a) Consider the Markov chain, X_n , on the states $i = 0, 1, 2, \dots$ with transition matrix

$$\begin{aligned} p_{i,i-1} &= p & i = 1, 2, \dots \\ p_{i,i+1} &= 1 - p & i = 0, 1, \dots \\ p_{0,0} &= p \end{aligned}$$

where $0 < p < 1$.

- (i) Show that the Markov chain is irreducible. [2 marks]
- (ii) Show that the Markov chain is aperiodic. [2 marks]
- (iii) Find a condition on p to make the Markov chain positive recurrent and find the stationary distribution in this case. [6 marks]
- (b) Consider the PageRank algorithm.
- (i) Describe PageRank as a Markov chain model for motion between nodes in a graph where the nodes correspond with web pages. [5 marks]
- (ii) Explain the main mathematical results that underpin the relevance of PageRank to a notion of web page “importance”. [5 marks]

END OF PAPER