THIS PAPER IS NOT TO BE REMOVED FROM THE EXAMINATION HALLS

UNIVERSITY OF LONDON **291 0226 ZB**

BSc Examination
for External Students

COMPUTING AND INFORMATION SYSTEMS AND CREATIVE COMPUTING

Software Engineering, Algorithm Design and Analysis

Dateline: Monday 18 May 2009 : 10.00 – 1.00 pm

Duration: 3 hours

Candidates should answer **FOUR** questions. Full marks will be awarded for complete answers to a total **FOUR** questions. Candidates must answer **TWO** questions from **Section A** and **TWO** questions from **Section B**.

Each question carries 25 marks. The marks for each part of a question are indicated at the end of the part in [ ] brackets.

There are 100 marks available on this paper.

No calculators may be used.

Section A

Question 1 Software Development Models and Processes

A large company is considering bids from various firms for a customer-facing IT project One has proposed following a *waterfall model*, one an *iterative model* and another has proposed what they call an *Extreme Programming Solution*

(a) Briefly, define the following:

- (i) Waterfall model [2]
- (ii) Iterative model [2]
- (iii) Extreme Programming [2]

(b) What is the first phase of the waterfall model? [1]

(c) What is involved in this phase? [3]

(d) Briefly list the strengths and weaknesses of the waterfall model. You should aim for at least four points. [4]

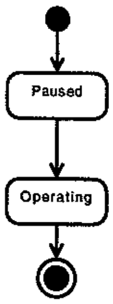
(e) Briefly list the advantages and disadvantages of an iterative model over a waterfall model You should aim for at least four points. [4]

(f) Is Extreme Programming iterative? [1]

(g) What is a software development model? [3]

(h) Why do software engineers follow development models rather than designing timetables and strategies from scratch for each client as the need arises? [3]

Question 2 State Machine Diagram



(a) The above state-machine diagram represents the main states of a washing machine. When it is first switched on, it is in a paused state, and then pressing the start button activates it.

(i) What is a pseudostate? Which pseudostates are represented in the diagram? [2]

(ii) The `Operating` state consists of three substates: `Washing`, `Rinsing` and `Spinning`. Redraw the state diagram to show this. [2]

(iii) When the door is opened in mid-cycle, reclosing it and pressing the start button should resume the cycle from where it left off. Show this in your diagram. What is the symbol used called? [2]

(iv) Add triggers for the transitions of your diagram. [2]

(v) The washing machine door will only open when the programme has not locked it. Show this on your diagram [2]

(vi) When the cycle finishes, the washing machine beeps. Show this on your diagram [2]

(b) Whirlpool^{TM}are developing a 'smart' washing machine that can be monitored and controlled by computers and mobile phones. An example of how this works is that, if you put on a wash whilst watching television, an alert message is displayed on the screen when the wash is finished and allows you to start a drying cycle running via the television set or to go and remove the wash load.

(i) What actors would be involved in such an interaction? [2]

(ii) Write a use case illustrating this alert functionality. [5]

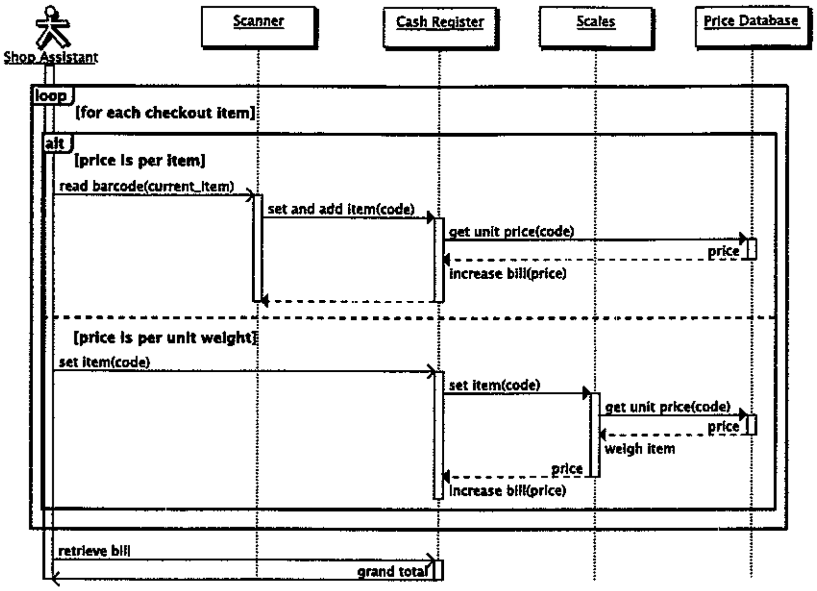
(iii) If the door was not properly closed when the cycle was started, a message is displayed on the washing machine screen. If nothing is done for 5 minutes, an alert is delivered to the television. Add an extension to your use case to illustrate this [2]

(iv) Whirlpool conducted testing by setting the system up in a number of households in Atlanta. Was this verification or validation? Explain your answer. [2]

(v) What challenges would Whirlpool's software engineers have faced testing just the software components of their system in the new washing machine? How might they have approached the problem? [2]

Question 3 Sequence and Class diagrams

The following sequence diagram shows the process of operating a supermarket checkout for a given customer



(a) (i) Why is the actor in this diagram not the customer? [1]

(ii) Explain the behaviour of `loop` and its guard in the diagram. [2]

(iii) Explain the behaviour of `alt` and its guards in the diagram. [2]

(b) (i) What classes are involved in this diagram? [1]

(ii) What class or classes interact with the `price database`? [1]

(iii) Draw a class diagram showing the details of the classes depicted in the sequence diagram. [5]

(c) When they pay for their goods, customers may present a loyalty card, which keeps a record of each customer's purchases. The system implementing this includes a class for each of `CustomerRecord`, for `Transaction` and for `Product`.

(i) Draw a simple class diagram showing these three classes and the relationship between them (include *multiplicities* in your diagram). [3]

(ii) Draw an activity diagram to show the shop assistant taking payment for a transaction and then—if the customer has a loyalty card—adding the transaction to their record. [6]

(d) (i) What aspects of the system might you subject to stress testing? [1]

(ii) What other performance testing might you carry out? [1]

(iii) The barcode scanner is only subjected to *black box testing*. What does this mean, and why might this be the case? [2]

Section B

Question 4

(a) Explain in one sentence the term *computational complexity* in the context of Algorithms. Discuss briefly the time complexity in the *worst* case for the algorithm below. Indicate the basic operations you have counted. Assume that i, j and n are positive integers. [5]

```
$i \leftarrow n, j \leftarrow n$
while $j \geq 1$ do
    for $i \leftarrow 1; i < j; i \leftarrow i + 1$ do
        $x \leftarrow x + 1$
    end for
    $j \leftarrow j$ div 3
end while
```

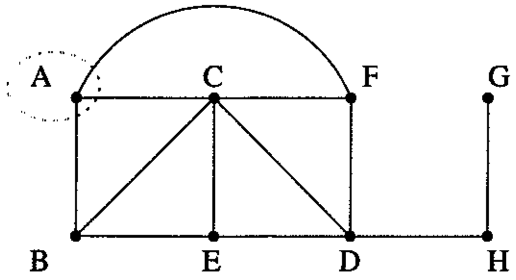
(b) Draw the binary search tree for the sequence of characters (c,a,e,k,m,g,b), where the alphabet letters are inserted in the given order. Demonstrate a pre-order traversal of the binary search tree. [4]

(c) Write a recursive algorithm for a pre-order traversal of a binary tree. [3]

(d) Would it be possible for the above binary tree structure to be preserved in an array? If yes, explain and demonstrate how. If no, explain why not. [3]

(e) Explain, with the aid of an example, in what situations the divide and conquer approach may be inefficient. Justify your answer. [5]

(f) Consider the connected graph below. Starting from vertex **A**, write the vertex sequence in the order that each vertex is visited applying the *depth first* traversal algorithm. Justify your answer. [5]



Question 5

(a) What is it meant by a polynomial time algorithm? Analyse the algorithm below. Explain why the algorithm is not normally considered as a polynomial time algorithm in the context of efficient algorithms. [5]

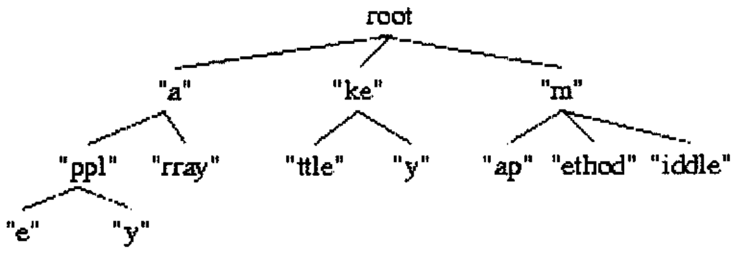
```
int algorithmX(n)

result ← 2
for $i$ ← 1 to $n$ do
    result ← result * result
end for
return result
```

(b) Outline the Boyer-Moore pattern matching algorithm, with an example where the text T and the Pattern P are as below. Demonstrate step by step the shifts and comparisons performed by the algorithm. [10]

```
T: a b a c a a b a d c a b a c a b a a b b
P: a b a c a b
```

(c) Consider the diagram below. Name the data structure and list all the strings represented in the structure. [5]



(d) Write an algorithm to compute the degree of each vertex of a simple graph. The graph is represented by its adjacency matrix `adjMatrix[i,j]`, where $i = 1, \cdots n$, $j = 1, \cdots n$ and n is the number of vertices of the graph. [5]

Question 6

(a) Consider a task of packing n objects into a minimum number of bins. To simplify the problem, we assume that everything is two dimensional, i.e. bins and objects are merely rectangles. The height of the n objects are $h_1, h_2, \cdots, h_n$. The width of the objects are all the same in size and can just be fitted in the rectangluar bins. The available bins are all of capacity C. [12]

(i) Draw a diagram to show one instance of the problem.
(ii) Classify the problem based on your algorithmic knowledge.
(iii) How 'easy' is this problem from algorithmic point of view?
(iv) What is the so-called "greedy approach"?

(b) Following the above, design and propose an approximation algorithm for the bin-packing problem using a greedy approach. Show all your work and highlight (or mark clearly) the final version of the algorithm. [13]

Hint: You may like to first sort in descending order the objects according to their sizes. Then take a so-called first-fit strategy placing an object in the first bin in which it fits, and starting a new bin if it does not fit in.

END OF EXAMINATION