UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2001

BEng Honours Degree in Computing Part III
BEng Honours Degree in Information Systems Engineering Part III
MEng Honours Degree in Information Systems Engineering Part III
BSc Honours Degree in Mathematics and Computer Science Part III
MSci Honours Degree in Mathematics and Computer Science Part III
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Associateship of the City and Guilds of London Institute
This paper is also taken for the relevant examinations for the
Associateship of the Royal College of Science

PAPER C302=I3.8

SOFTWARE ENGINEERING - METHODS

Tuesday 8 May 2001, 10:00
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators required

Part A – *please answer in separate answer booklet*

1 Consider a simple language with: assignments, **A**, ; binary conditionals, **C(_,_)** ; loops, **L(_)** ; and binary sequencing, **S(_,_)** .

a For this language, define hierarchically a measure which gives the total number of paths through a piece of code where loops are executed at most twice (i.e. "All Paths" with k = 2).

b Give the Syntax Tree and Control Flowgraph for the following program fragment and use the above measure to evaluate the number of paths through it (again with loops executed at most twice).

```
1. while (x<10) do
2.    ( x := x+1 ;
3.      while (y < x) do
4.         (y := y+1
           )
      )
5.    stop
```

c List all of the paths from part b and, for the longest, use Symbolic Execution to find conditions on the inputs which must be satisfied to execute it. Explain whether it is possible to achieve 100% coverage for this program with this strategy and, if not, state what is the maximum coverage possible.

The three parts carry, respectively, 20%, 30% , 50% of the marks.

2a Describe briefly what kinds of testing are most appropriate at the unit test, integration test and system test stages of the software lifecycle. What kinds of error are being looked for at each stage?

b In white box testing, for test case selection, what distinguishes:

i) data flow based techniques,

ii) adaptive techniques.

What advantages are claimed for each?
What assumptions underlie all forms of white-box software testing?

c Explain the terms *on point*, *off point* and *extreme point* as used in boundary value analysis. Give an example of each of these for the region bounded by x>0, x≤1, y>0, y≤1 for real valued x and y.

d Briefly describe how probabilistic techniques can be used to determine the reliability of software.

The four parts carry, respectively, 30%, 30% , 30%, 10% of the marks.

Part B – *please answer in separate answer booklet*

3a Very briefly explain what a Design Pattern is.

b Define what is meant by the State Design Pattern and explain what its benefits are.

c A class `Barrier` has a public method `open()` which returns true when the barrier is up and false otherwise. The class also exports the method `changeState(State s)` which changes the state of the barrier to `Up`, `Down`, `Raising` and `Lowering`. Using the State Design Pattern to implement `Barrier:`

i) Draw a class diagram showing the relationships between the classes `Barrier`, `State`, `Up`, `Down,` `Raising` and `Lowering`.

ii) Give an implementation in Java for each of the classes `Barrier`, `State`, `Up`, `Down,` `Raising` and `Lowering`.

d Suggest a means of ensuring that only one instance of each state object exists and demonstrate how this might be done by modifying the code for one of the classes from c ii).

The four parts carry, respectively, 10%, 20% , 50%, 20% of the marks.

4a Name and briefly define the four main elements of a Design Pattern description.

b Briefly describe the purpose of the Composite Design Pattern and draw a class diagram showing the relationships between its participants.

c A file system in a computer system is managed using two sorts of objects: `Directory` and `File`. Each file object records the size in bytes of the file it manages. The size returned by a directory is the sum of the sizes of the files and directories it contains. Assuming the use of Composite, identify the classes required to represent the file system and give an implementation in Java for each of these classes. Include a method `size()`, which computes size in bytes. Use a constant to represent the size of files.

d Assuming that each file and directory has a name, sketch the implementation of a method `String pathname()` which returns the global name of a file or directory (e.g. file *fred* in directory *jim* within parent directory *bill* has the global name "*bill/jim/fred.*"). Assume each class implements a method `name()` which returns the name of the file or directory. Which pattern have you used to implement `pathname()`?

The four parts carry, respectively, 10%, 20% , 40%, 30% of the marks.