

1997 Paper 3 Question 3

Further Modula-3

The thread system in Modula-3 uses mutexes and condition variables to control concurrency. An alternative scheme would be to use message passing.

Each message-passing channel would be an object with two methods:

```
send (message: REFANY)
```

and

```
receive (): REFANY
```

plus an initialisation method.

Calling `send` would append the argument to a (fixed-length) FIFO queue of data which would be collected by the thread calling `receive`. The thread would block if `receive` were called when the queue was empty or if `send` were called when the buffer was full.

Write an interface, `Message`, defining a suitable opaque object type with `init` and `send` methods, and write a further interface `MessageInternal` revealing the `receive` method. [8 marks]

Write an implementation exporting to both of these interfaces supplying the concrete revelations of the types and providing appropriate default methods. [12 marks]