

## 1997 Paper 10 Question 2

### Modula-3

The following code has been extracted from a Modula-3 program written to maintain information about the collection of bicycles and tricycles in a transport museum:

```
TYPE
  Cycle = OBJECT
    year   : CARDINAL;
    weight : REAL
  METHODS
    info () : TEXT      := NoInfo
  END;

  TriCycle = Cycle OBJECT
    dl : REAL;
    df : REAL;
    dr : REAL
  OVERRIDES
    info      := Tinfo
  END;

PROCEDURE Tinfo (c : TriCycle) : TEXT =
  BEGIN
    RETURN "Year: " & Fmt.Int(c.year) &
           " Weight: " & Fmt.Real(c.weight) &
           " Mean Diameter: " & Fmt.Real((c.dl+c.df+c.dr)/3.0)
  END Tinfo;

VAR
  rudge := NEW (TriCycle, year := 1927, weight := 30.6,
               dl := 0.55, df := 0.66, dr := 0.77);
```

Supply a suitable procedure `NoInfo` and then explain the code when this fragment is augmented by your new procedure. The explanation should particularly describe the object `rudge`. [8 marks]

Explain the operation and effects of each of the following `IO.Put` statements:

```
IO.Put (rudge.info () & "\n");
IO.Put (TriCycle.info (rudge) & "\n");
IO.Put (Cycle.info (rudge) & "\n");
IO.Put (NARROW (rudge, Cycle).info () & "\n");
```

[12 marks]