Modula-3

The following is a fragment of a Modula-3 program which is supplied with data arranged as one left-adjusted integer per line. The integers (which may be assumed to be all different) are read one at a time and the procedure Put assembles a simple binary tree by arranging for each integer in turn to be in a new node of the tree. The assembly process sorts the integers into numerical order.

VAR
  tree : PtrToNode := NIL;
BEGIN
  TRY
    LOOP
      TRY
        Put (Scan.Int (Rd.GetLine (Stdio.stdin)), tree)
      EXCEPT
        Scan.BadFormat =>
          Wr.PutText (Stdio.stdout, "Bad datum\n")
      END
      END
    EXCEPT
      Rd.EndOfFile =>
        END;
      PrintTree (tree);
      Wr.Close (Stdio.stdout)
    END
  END
END

Explain the operation of the two TRY–EXCEPT clauses. [5 marks]

Provide a suitable TYPE statement to define the type PtrToNode. [5 marks]

Write a procedure Put and explain its operation. [5 marks]

The procedure PrintTree is intended to write out the values in the tree in ascending order. Write this procedure and explain its operation. [5 marks]