ALGORITHM 44
BESSEL FUNCTIONS COMPUTED RECURSIVELY
MARTIN E. WOJCIECH
RCA Digital Computation and Simulation Group,
Mooresville, New Jersey

procedure Bessr(N, FX, LX, Z) Result: (J, Y);
  value LX, FX, N;
  real FX, LX, Z; real array J, Y; integer N;
comment Bessel Functions of the first and second kind, \( J_\nu(X) \) and \( Y_\nu(X) \), integral order \( \nu \), are computed by recursion for values of \( X \), \( FX \leq X \leq LX \), in steps of \( Z \). The functions are computed for values of \( \nu \), \( 0 \leq \nu \leq N \). M[SUB], the initial value of \( \nu \) being chosen according to formulae in Erdélyi's *Asymptotic Expansions*. The computed values of \( J_\nu(X) \) and \( Y_\nu(X) \) are stored as column vectors for constant argument in matrices \( J \), \( Y \) of dimension \((N+1) \times 1 \) by entier \((LX - FX)/Z + 1) \);
begin real PI, X, GAMMA, PAR, LAMDA, SUM, SUM1;
  integer P, SUB, MAXSUB;
  PI := 3.14159265;
  GAMMA := .57721566;
  PAR := 63.0 - 1.5 \times \ln (2 \times PI);
  MAXSUB := entier ((LX - FX)/Z);
begin real array JHAT [0:N, 0:MAXSUB];
  integer array M[0:MAXSUB];
  SUB := 0;
  for X := FX step Z until LX do
    begin if \((X > 0) \land (X < 10)\) then M[SUB] := 2 \times entier \((X) + 9\)
    else
      begin real ALOG;
        ALOG := (PAR - 1.5 \times \ln (X))/X;
        M[SUB] := entier \((X \times (exp (ALOG)) \times \exp \((ALOG)/2\))\);
      end;
      if N > M[SUB] then
    begin for P := M[SUB] + 1 step 1 until N do
      J [P, SUB] := 0 end;
      JHAT [M[SUB], SUB] := 10 \uparrow (-9);
    comment Having set the uppermost \( J_\nu(X) \) to a very small number we are now going to compute all the \( J_\nu(X) \) down to \( P = 0 \);
    for P := M[SUB] step -1 until 1 do
      JHAT [P-1, SUB] := 2 \times P/X \times JHAT [P, SUB] - JHAT [P+1, SUB];
      SUM := SUM1 := 0;
  for P := 2 step 2 until \((M[SUB] + 2)\) do
    SUM := SUM + JHAT [P, SUB];
    LAMDA := JHAT [0, SUB] + 2 \times SUM;
  for P := 0 step 1 until N do
  comment \( J_\nu(X) \) have been computed by use of \( \tilde{J}_\nu(X) \);
  for P := 2 step 2 until \((M[SUB] + 2)\) do
    SUM1 := SUM1 + \((-1) \times \downarrow \uparrow \uparrow \) \((2 \times P, SUB)\)
    \(\times \downarrow \uparrow \downarrow \) \(\downarrow \downarrow \);\n    \(Y[0, SUB] := 2/PI \times (J[0, SUB] \times (GAMMA + \ln(X/2))) + 4 \times \downarrow \uparrow \downarrow \);\n  for P := 0 step 1 until \((M[SUB]-1)\) do
  SUB := SUB + 1 end end end