```
program CoroutinesTowersHanoi;
(* towers of hanoi with coroutines *)

(* there are three towers built of decreasing rings stringed onto sticks    *)
(* at the initial state all rings are stringed onto stick no. 1. our job is *)
(* to move all rings from the stick 1 to the stick 3. the difficulty is     *)
(* that we mustn't violate the following conditions                         *)
(*    1. we can move only one ring at one step                              *)
(*    2. each ring may be placed only onto a greater one                    *)
(* to manage with this difficult problem we have an auxilliary stick 2      *)

 unit WZ:coroutine(N,F,T:integer);
   (* move n rings from stick f to stick t *)
      var K:integer;
 begin
   return;
   do
      K:=6-(F+T);
      if N>1 then  attach (P(N-1,F,K)); fi;
      call MODYF(F,T);  (* MOVE ONLY ONE RING *)
      if N>1 then attach (P(N-1,K,T)); fi;
      detach;
   od;
 end WZ;

unit MODYF:procedure(F,T:INTEGER);
   (* move the topmost ring from stick f to stick t *)
begin
   TOP(T):=TOP(T)+1;
   W(T,TOP(T)):=W(F,TOP(F));
   W(F,TOP(F)):=0;
   TOP(F):=TOP(F)-1;
   call DISPL;
end MODYF;

unit DISPL:procedure;
  (* PRINTING *)
  var T,I,J,K,M,N:INTEGER;
begin
   T:=1;
   for I:=2 to 3 do
    if TOP(I)>TOP(T) then T:=I fi
   od;
   T:=TOP(T);
   for I:=T downto 1 do
      M:=15;
      for J:=1 to 3 do
        for K:=1 to M do WRITE(" "); od;
        if W(J,I)=/=0 then
          for K:=1 to W(J,I) do WRITE("*") od;
        fi;
        M:=15-W(J,I);
      od;
    WRITELN;
   od;
   for I:=1 to 15 do WRITE(" "); od;
   for I:=1 to 45 do WRITE("-"); od;
   WRITELN;
end DISPL;
```

```
 var  W:arrayof arrayof INTEGER,   (* how many rings are stringed *)
                                   (* on each stick               *)
      TOP:arrayof INTEGER,  (* the topmost ring size on each stick *)
      NB,I,J,K,TIMEB:INTEGER,
      P:arrayof arrayof arrayof WZ; (* coroutine pointers *)

    begin
     array W dim(1:3);
     array TOP dim(1:3);
     WRITELN(" PROGRAM TOWERS OF HANOI");
     WRITELN(" VERSION WITH COROUTINES");
     do WRITELN(" GIVE THE NUMBER OF RINGS");
        READ(NB);
        WRITELN(NB);
        if NB>0 then exit
        else
          WRITELN(" NUMBER OF RINGS MUST BE GREATER THAN 0")
        fi
     od;
     TIMEB:=TIME;
     TOP(1):=NB;
     array W(1) dim(1:NB);
     array W(2) dim(1:NB);
     array W(3) dim(1:NB);
     K:=NB;
     for I:=1 to NB do
       W(1,I):=K;
       K:=K-1;
     od;
     (* STICK 1 IS FULL *)
     WRITELN(" THE ALGORITHM ACTS AS FOLLOWS");
     call DISPL;
     array P dim (1:NB);
     for I:=1 to NB
     do array P(I) dim(1:3);
       for J:=1 to 3
       do array P(I,J) dim(1:3);
          for K:=1 to 3
          do
            if J=/=K then P(I,J,K):=new WZ(I,J,K) fi
          od
       od
     od;
     attach (P(NB,1,3));
     WRITELN(" EXECUTION TIME FOR",NB:4," RINGS =",TIME-TIMEB," SEC");
end
```