```
Unit Unt_iopcc;    {$mode       delphi}{$H+}       Interface     Procedure     Main;
Implementation   {$define _}   type L=integer     ;I=0..3;const   Z=1 shl 11;    U=$13
shl        1+1;  O=U     and   -U;D               =O shl          (O shl    O+    O);C=
' _|'      ;B=  $597B   ;l0:   array             [I]of           L=(0,    -3,U   -O,-O)
;lQ:array[I]    of L=(O,U,-O,  -U);var E:array    [0..Z]         of L;function    H(var
Q:L):L;begin    H:=Q; Inc (Q);     end;procedure  Q(U:L);        begin if U < D   then
Write           (C[B      shr               (H(U) shl 1)         and       3]+C   [B shr
( U              shl      1)                and 3  ])else       Write      (Copy  (C+C+C+LineEnding,
U+O,             3))     ;end;       Procedure      Main;var    ll:        array  [ 0 .. Z ] of L;
// #             This     is       Just an Du-       mmy Com-   ent to      fill  the Gaps ********


                                    Hel:l=0;
                                    Wor:l=d-D;

                                    P,A,S,CA:L;
                                  _fe:l=z-z;_C,oo:l;


     l1:array           [I]of L;begin             Randomize;           E[0]:=
  d;CA:=U*U-O;A       :=CA;E[CA]:=Z+2;          while(Wor<>0)        or(_fe
  >=Hel)      do     begin       oO:=0        ;S:=CA      ;CA:=       A;_C:=
  E[S];              Wor:=0      ;for P       in l0       do          begin
  A:=lQ              [P and      3]+S;        if((A       >=0)        and(A<
  U*U)               and(P<>     (A mod       U))and      ((_C        and Z)
  <>(E               [A]and       Z)))        then        begin       l1[ H(
  Wor)               ]:=P;        end;        end ;        if(        Wor<>0
  )then              begin        P:=l1[      Random     (Wor)]       and 3;
  A:=lQ              [P]+S        ;E[S]       :=_C        or O        shl P;
  E[A]               :=E[A]or     Z or(       O shl(      (P+2)       mod 4));ll[H(_fe)]
  :=A;end             else         begin       if _fe     >=Hel       then A:=ll[H(Hel)];
  end;end      ;      Q(d-2);     for S       :=O to     U-O do       Q(D shr O);Q(D+O);
   for S:=O to U       do  begin  for            A:=O to U do         Q(E[H(Oo)]and 6
    );Q(D);end;          {$IFDEF _}             Readln;             {$ENDIF}End;end.
```