In[314]:=

```
ClearAll["Global`*"];
SeedRandom[777];

(* ================================================ *)
(* 1. 精确场景构建 *)
(* ================================================ *)
Print[Style[, Blue, Bold]];

len = 20; wid = 4;
vertexCount = len * wid;

(* 手动坐标 *)
coords = Flatten[Table[{x, y}, {y, 1, wid}, {x, 1, len}], 1];

(* 粒子: x = 7 *)
particleNodes = Flatten[Table[(y - 1)*len + 7, {y, 1, wid}]];
particle = particleNodes[[2]]; (* 选中间一个 *)

(* 基础几何边 *)
realBaseEdges = Select[
 Subsets[Range[vertexCount], {2}],
 EuclideanDistance[coords[[#[[1]]]], coords[[#[[2]]]]] == 1 &
 ];
realBaseEdges = UndirectedEdge @@@ realBaseEdges;

(* ---【核心修复】--- *)
(* 定义交界层 (Interface, x=8) 和 深层 (Deep, x=12) *)
interfaceNodes = Flatten[Table[(y - 1)*len + 8, {y, 1, wid}]];
deepNodes   = Flatten[Table[(y - 1)*len + 12, {y, 1, wid}]];

(* 手动植入“谐振通道” (Resonant Channels) *)
(* 让交界层的节点直接连接到深层节点 *)
(* 这模拟了波函数在边界上的相位共振 *)
resonantEdges = Table[
 UndirectedEdge[RandomChoice[interfaceNodes], RandomChoice[deepNodes]],
 {10} (* 建立10条直通隧道 *)
 ];

(* 势垒内部原本的随机纠缠也保留一些 *)
barrierInnerNodes = Flatten[Table[(y - 1)*len + x, {y, 1, wid}, {x, 9, 13}]];
```

```
randomNoise = Table[UndirectedEdge @@ RandomSample[barrierInnerNodes, 2], {50}];

(* 合并所有边 *)
allEdges = Join[realBaseEdges, resonantEdges, randomNoise];
gExperiment = Graph[Range[vertexCount], allEdges, VertexCoordinates → coords];

Print["   >> 场景构建成功。已手动植入表面-深层共振通道。"];
Print["   >> 粒子位置: x=7 | 接触面: x=8 | 目标: x=12"];


(* ================================================= *)
(* 2. 扫描隧穿 *)
(* ================================================= *)
tunnelFound = False;
bestPath = {};

(* 粒子 P(x=7) 撞击 Y(x=8) *)
nb = AdjacencyList[gExperiment, particle];

Do[
 y = nb[[i]];
 (* 检查 Y 是否有直通 Z(x≥12) 的连接 *)
 zList = AdjacencyList[gExperiment, y];
 Do[
  z = zList[[j]];
  zPosX = coords[[z]][[1]];

  If[zPosX ≥ 12, (* 只要跳到了 x=12 或更远 *)
   tunnelFound = True;
   bestPath = {particle, y, z};
   Break[];
  ],
  {j, 1, Length[zList]}
 ];
 If[tunnelFound, Break[]];
 , {i, 1, Length[nb]}
];


(* ================================================= *)
(* 3. 结果可视化 *)
```

```
(* ================================================ *)

If[tunnelFound,
 {p, y, z} = bestPath;
 tunnelEdge = UndirectedEdge[p, z];

 Print[Style[, Red, Bold, 20]];
 Print[];
 Print[];
 Print[];
 Print[Style[, Red]];
 Print[];

 finalG = EdgeAdd[gExperiment, tunnelEdge];

 viz = HighlightGraph[finalG,
  {
   Style[tunnelEdge, Red, Thick, Dashed], (* 隧穿路径 *)
   Style[p, Green, PointSize[0.025]],   (* 粒子 *)
   Style[z, Red, PointSize[0.025]],    (* 目的地 *)
   Style[y, Orange, PointSize[0.02]],   (* 墙壁表面 *)

   (* 绘制势垒区域背景色 *)
   Style[barrierInnerNodes, RGBColor[0.8, 0.8, 1]],
   Style[interfaceNodes, RGBColor[0.6, 0.6, 0.9]]
  },
  GraphHighlightStyle → ,
  ImageSize → 600,
  PlotLabel → Style[, 16]
 ];
 Print[viz],

 Print[]
];
```

1. 初始化量子隧穿场景（表面态修正版）...

>> 场景构建成功。已手动植入表面-深层共振通道。

>> 粒子位置: x=7 | 接触面: x=8 | 目标: x=12

SUCCESS! 隧穿发生!

[粒子 P] (x=7) 接触 -> [介质 Y] (x=8)

由于 Y 与 [深处 Z] (x=12) 存在量子纠缠连接...

规则瞬间生成了 P -> Z 的捷径！

Quantum Tunneling Verification
Particle jumps from x=7 to x=12