EXERCISE 5:
IPR dependence on the basis

XX model

(* Parameters of the Hamiltonian *)
Clear[chainsize, upspins, downspins, dim, Jxy, Jz, open];
chainsize = 10;
upspins = chainsize / 2;
downspins = chainsize - upspins;
dim = chainsize! / (upspins! downspins!);
Jxy = 1.0;
Jz = 0.0;
open = 1;

(* Creating the basis *)
Clear[onebasisvector, basis];
onebasisvector = Flatten[{Table[1, {k, 1, upspins}], Table[0, {k, 1, downspins}] tutti]};
basis = Permutations[onebasisvector];

(* ELEMENTS OF THE HAMILTONIAN *)

(* Initialization *)
Clear[HH];
Do[Do[HH[i, j] = 0., {j, 1, dim}], {i, 1, dim}];

(* Diagonal elements - Ising interaction *)
Do[
   Do[
      HH[i, i] = HH[i, i] + (Jz / 4.) * (-1.)^(basis[[i, k]] + basis[[i, k + 1]]);
      , {k, 1, chainsize - 1}];
      , {i, 1, dim}];

(* Term included in the Ising interaction if the chain is closed *)
If[open == 0, Do[
   HH[i, i] = HH[i, i] + (Jz / 4.) * (-1.)^(basis[[i, chainsize]] + basis[[i, 1]]),
   {i, 1, dim}];

(* Off-diagonal elements - flip-flop term *)
Clear[howmany, site];
Do[
   (* Initialization *)
   howmany = 0;
   Do[site[z] = 0, {z, 1, chainsize}];
   (* Sites where states i and j differ *)
   Do[If[basis[[i, k]] != basis[[j, k]], {howmany = howmany + 1, site[howmany] = k}],
      {k, 1, chainsize}];
   (* Coupling matrix element - when only two neighbor sites differ *)
   If[howmany == 2,
      If[site[2] - site[1] == 1,
      (* Additional term for closed system *)
      If[open == 0, If[site[2] - site[1] == chainsize - 1,
         , {i, 1, dim}]];
   (* Hamiltonian *)
   Clear[HamiltonianXX];
   HamiltonianXX = Table[Table[HH[i, j], {i, 1, dim}], {j, 1, dim}];
   (* Diagonalization *)
   Clear[EnergyXX, VectorXX];
   EnergyXX = Chop[Eigenvalues[HamiltonianXX]];
   VectorXX = Chop[Eigenvectors[HamiltonianXX]]];
XXZ model

(* Parameters of the Hamiltonian *)
Clear[chainsize, upspins, downspins, dim, Jxy, open, total];
chainsize = 10;
upspins = chainsize/2;
downspins = chainsize - upspins;
dim = chainsize!/(upspins! downspins!);
Jxy = 1.0;
open = 1;
total = 51;

(* Loop for values of Jz *)
Do[
  Jz = 0.5 (kk - 1);

  (* ELEMENTS OF THE HAMILTONIAN *)
  (* Initialization *)
  Clear[HH];
  Do[Do[HH[i, j] = 0., {j, 1, dim}], {i, 1, dim}];

  (* Diagonal elements - Ising interaction *)
  Do[
    Do[
      HH[i, i] = HH[i, i] + (Jz/4.) * (-1.)^basis[[i, k]] + basis[[i, k+1]]);
      , {k, 1, chainsize-1};
    , {i, 1, dim}];

  (* Term included in the Ising interaction if the chain is closed *)
  If[open == 0, Do[
    HH[i, i] =
    HH[i, i] + (Jz/4.) * (-1.)^basis[i, chainsize] + basis[i, 1]],[i, 1, dim]];

  (* Off-diagonal elements - flip-flop term *)
  Clear[howmany, site];
  Do[
    Do[
      (* Initialization *)
      howmany = 0;
      Do[site[z] = 0, {z, 1, chainsize}];
      (* Sites where states i and j differ *)
      Do[If[basis[i, k] != basis[i, k], {howmany = howmany + 1, site[howmany] = k}],
        {k, 1, chainsize}];
      (* Coupling matrix element - when only two neighbor sites differ *)
      If[howmany == 2,
        If[site[2] - site[1] == 1,

      (* Additional term for closed system *)
      If[open == 0, If[site[2] - site[1] == chainsize - 1,

    , {j, 1 + dim}];
    , {i, 1, dim - 1}];

  (* Hamiltonian *)
  Clear[Hamiltonian];
  Hamiltonian = Table[Table[HH[i, j], {i, 1, dim}], {j, 1, dim}];

  (* Diagonalization *)
  Clear[Energy, Vector];
  Energy = Chop[Eigenvalues[Hamiltonian]]; Vector = Chop[ EVPectors[Hamiltonian] ];

  (* CHANGE to the XX BASIS *)
  (* NOTICE that in FORTRAN, the eigenstates are columns while in Mathematica they are rows!! *)
  (* In Fortran the transformation would be TRANSPOSE[VecXX].Vector *)
Clear[Vectornew];
Vectornew = VectorXX.Transpose[Vector];

(* Inverse Participation Ratio in XX BASIS *)
Clear[IPRxx];
IPRxx = 0.0;
Do[
  Clear[denom];
  denom = Sum[Vectornew[[i, k]]^4, {k, 1, dim}];
  IPRxx = IPRxx + 1 / denom;
  , {i, 1, dim}];

(* Average value of IPR *)
AveIPRxx[kk] = IPRxx / dim;
Print[{Jz, AveIPRxx[kk]}];
, {kk, 1, total}];

Plot
Blue: IPR in the site-basis
Red: IPR in the XX basis

(* Plot: IPR vs Jz *)
Clear[tab, tabnew];
tab = Table[{0.5 (kk-1), AveIPR[kk]}, {kk, 1, total}];
tabnew = Table[{0.5 (kk-1), AveIPRxx[kk]}, {kk, 1, total}];
ListPlot[{tab, tabnew}, Joined -> True,
  PlotRange -> All, PlotStyle -> {{Thick, Blue}, {Thick, Red}},
  LabelStyle -> Directive[Black, Bold, Medium], AxesLabel -> {"Jz", "<IPR>"}]

<IPR>