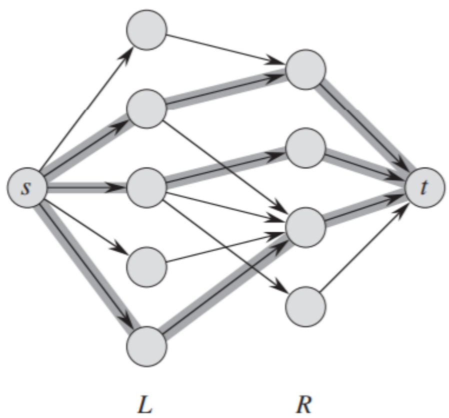
Question 2

For this $n \times n$ chessboard, a 2D array $Pos[][]$ could be established to indicate the valid position on the chessboard. Marking the position (i, j) invalid if (i, j) is the coordinate of bishops or it is located at the diagonal position of bishops. Otherwise marking (i, j) valid.

Then, defining that L is the set of vertexes that represent all the columns and R is the set of vertexes that represent all rows. To obtain a bipartite graph, each vertex in L is connected to every vertex in R to form all coordinate combinations. Before the connection, checking the position in $Pos[][]$. The two vertexes are connected only if the position that it represents in chessboard in valid. Otherwise skip to next vertex.

In such way, a bipartite graph $G = (V, E)$ is constructed. The maximum matching of the graph is the maximum number of rooks we can place. To achieve this, constructing a network flows $G' = (V', E')$ correspond to matchings. The capacity of G' is unit capacity 1 such that there is not two rooks at same position. Source s and sink t do not belong to V. G' could be shown as following diagram.



Then applying ***Ford-Fulkerson algorithm*** to find the maximum flow of G'. The number of edges from L to R in the maximum flow is the maximum number of rooks that we can place.