**Submitted by:**

**SANMESH GHONGADE**

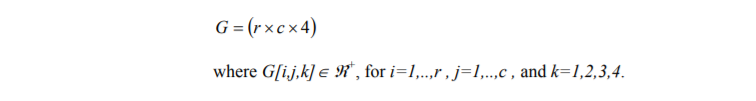
**1MS18SCS19**

**Jigsaw Puzzle**

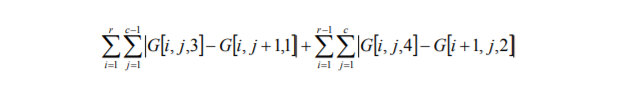
A jigsaw puzzle contains 500 pieces. A “section” of the puzzle is a set of one or more pieces that have been connected to each other. A “move” consists of connecting two sections. What is the minimum number of moves in which the puzzle can be completed?

**First Optimal Solution**

* The first step for solving jigsaw puzzles is to identify and separate the edge pieces from the rest of the puzzle.
* Search for the border pieces, then group elements into several piles of similar colour or markings. This makes the searching for a specific piece easier as focusing on fewer, similar pieces allows for better discrimination.
* The ease of recognition of boundary edges provides information about contiguous neighbours that reduces search spaces. It also defines the region in which other pieces need to be fitted.
* Pieces from previously created groups are then put in place to complete the puzzle by lining up the same shaped pieces with the same orientation and adjoining graphical content.
* A well trained eye will look for a combination of colour, markings and shape of the jigsaw puzzle piece as those features are the easiest to match up first.
* All other pieces with even the smallest amount of the colour or marking in question are usually fitted last. In this last stage of a construction process, shape often plays a more important role as many pieces are virtually indistinguishable by colour or markings alone.
* Canonical jigsaw puzzle pieces have four edges and can be rotated to one of four possible orientations.
* Puzzle pieces that can be contiguously assembled into a single entity are placed into a rectangular puzzle grid to form a complete picture that exactly fills the grid.
* Graphical and geometrical features of paired edges must be assessed to determine how well any two puzzle pieces fit together.
* Correctness of placement of puzzle pieces is measured by a function that reduces all the information contained in a puzzle piece to four single functional values, each of which is assigned to one of the four edges of the given piece. Let us represent a rectangular puzzle grid with c columns and r rows as an array of quadruples.



* Each element of this array represents a value associated with one of the edges of a puzzle piece. Let us denote values associated with left, top, right, and bottom edge of the piece positioned at (i,j) in the puzzle grid by G[i,j,1], Gl[i,j,2], Gl[i,j,3], Gl[i,j,4], respectively.
* The optimum solution of the jigsaw puzzle problem thus becomes the minimization of the sum.



* This sum can also be used in implementation of a cost function for improvement techniques requiring assessment of partial solutions. Assessment and improvement of partial solutions is particularly important for difficult variations of the jigsaw puzzle problem including incomplete jigsaw puzzles (puzzles with missing pieces) and puzzles containing unrelated pieces. A successful minimization of the proposed function ensures solutions for all canonical jigsaw puzzles regardless of their integrity as long as subsets of puzzle pieces do not need to be discarded to arrive at an optimal solution.

**Second Optimal Solution**

* The answer is 499 moves.
* The puzzle can be represented as a number of internal nodes in a tree therefore the answer 500-1=499
* Any move decreases the number of remaining sections by 1.
* Therefore, after *k* moves, the total number of remaining sections will be 500 − *k* irrespective of an order in which the sections are assembled. Hence 499 moves will be made before the entire puzzle is assembled.

**Generalised Solution**

* Any algorithm to assemble the puzzle can be represented by a binary tree whose leaves represents the single pieces and internal nodes represent connecting two sections (its children).
* Since each internal node in such a tree has two children, the leaves can be interpreted as extended nodes of a tree formed by its internal nodes.
* According to equality, the number of internal nodes (moves) is equal to n − 1, one less than the number of leaves (single pieces), for any such tree (assembling algorithm).
* Alternatively, one can reason that the task starts with n one-piece sections and ends with a single section.
* Since each move connects two sections and hence decreases the total number of sections by 1, the total number of moves made by any algorithm that doesn’t disconnect already connected pieces is equal to n − 1.