DESIGNER CO-PILOT: REIMAGINING WORKPLACE DESIGN WITH AI

Name : Jiyanshu

Roll no. - 220481

Mentor : Prof Niseeth Srivastav

IIT Kanpur | 2024-25

ABSTRACT

Our project focuses on designing workspaces by developing an AI-powered Designer Co-pilot that transitions traditional design tools into intelligent collaborators. We follow the grid philosophy, using structure layouts and spatial constraints to create efficient and organized workspace environments. At the core of this approach is the use of Integer Linear Programming (ILP) and different sampling methods to produce the layouts which best fits all the user-given constraints. The Co-pilot utilizes data-driven intelligence and generative abilities to actively participate in layout planning, compliance testing and decision making.

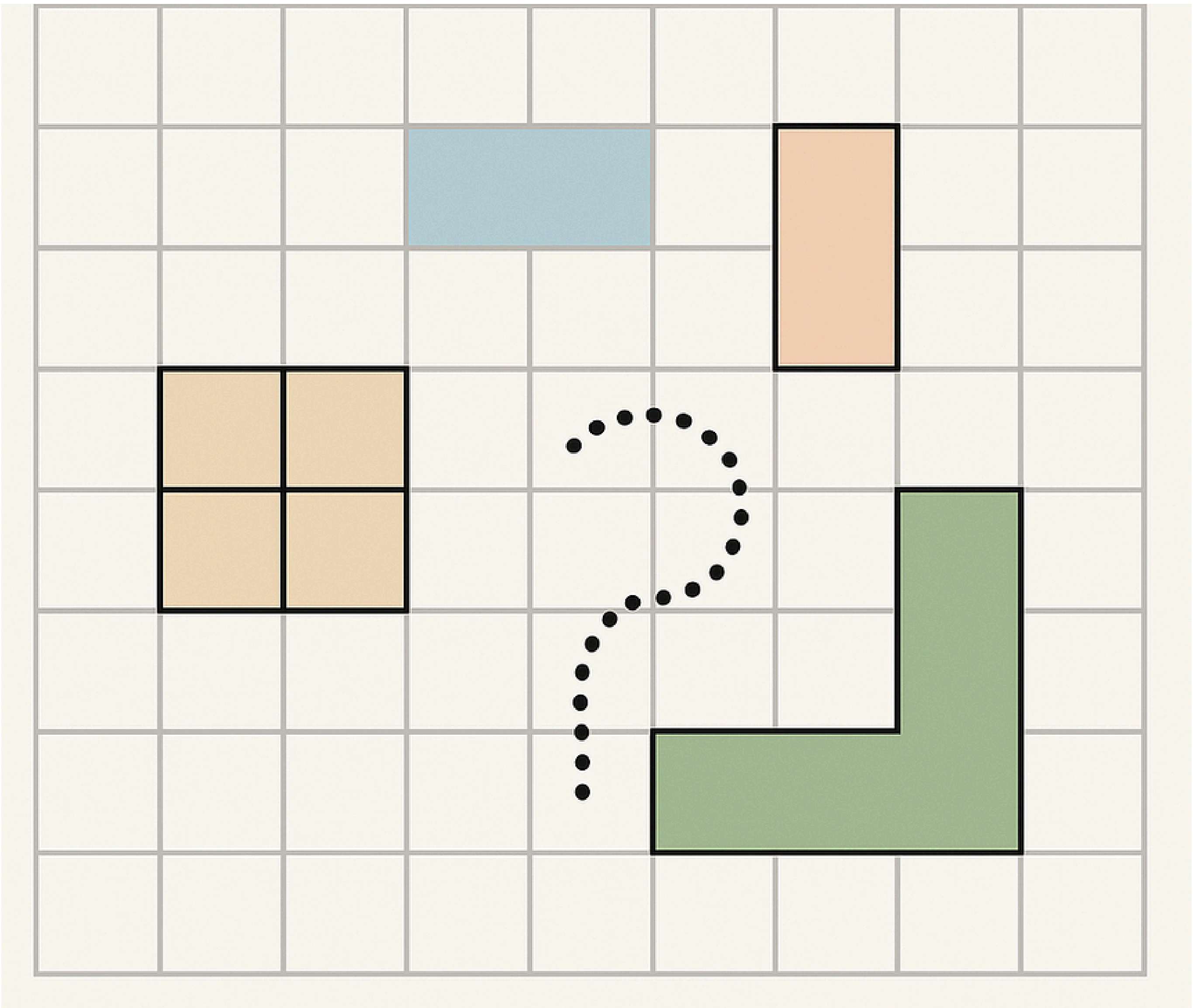
INTRODUCTION

Our project focuses on the development of a Designer Co-pilot—an AI-based tool that supports real-time decision-making, ensures regulatory compliance, and offers intelligent design suggestions to optimize spatial efficiency and user experience.

To implement our solution, we adopt the grid-based design philosophy combined with Integer Linear Programming (ILP) to model and solve spatial layout constraints effectively.

GRID PHILOSOPHY

The grid philosophy breaks down the workspace into uniform units or "cells", allowing spatial elements such as desks, rooms, and pathways to be represented in a structured and discrete format.



INTEGER LINEAR PROGRAMMING

ILP is mathematical optimization program in which some or all of the variables are restricted to be integers.

In many settings the objective function and the constraints (other than integer constraints) are linear.

$$\begin{aligned} \max\ & c^\top x \\ \text{s.t. } Ax &\le b \\ x &\ge 0 \\ x &\in \mathbb{Z}^n \end{aligned}$$

INTEGER LINEAR PROGRAMMING

ILP is a general method used to solve problems if we are able to express the constraints as linear equations.

Users often describe their constraints in logical terms (such as adjacency, patterns, etc.) or geometrical terms (such as periphery, diagonal positioning, etc.)

CONSTRAINTS

We will be analyzing and implementing various single constraints individually :

- Periphery
- Diagonal
- Adjacency
- Big Colored Block
- Pattern
- No two adjacent tiles can have the same color

PERIPHERY CONSTRAINT

Objective Function

$$\max \sum_{(i,j)\in P} \sum_{c\in\{R,G,B\}} w_c \cdot G^c_{i,j}$$

All the loud salespeople need to sit such that their chatter doesn't bother the quieter employees. What arrangement works best?

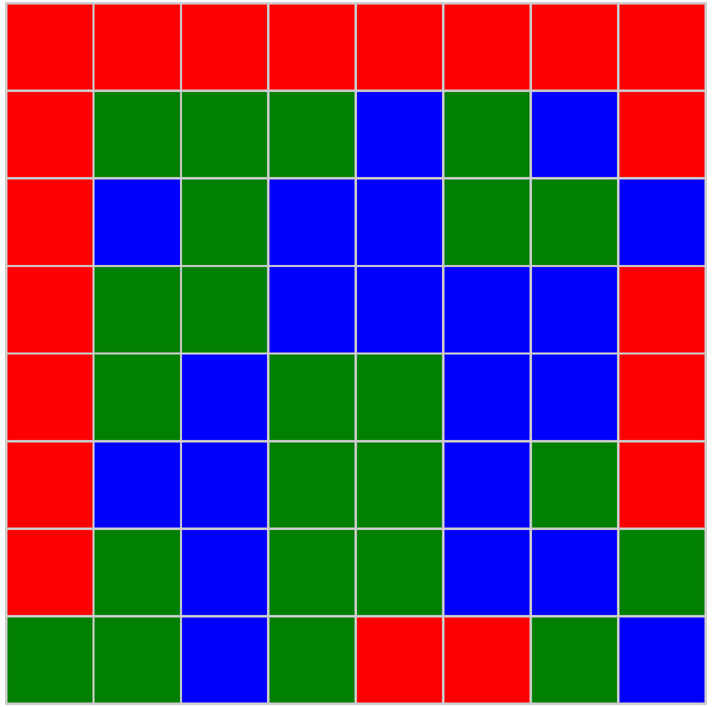
Choose the Periphery Constraint to place the salespeople along the edges of the workspace. This will minimize disruption to the quieter employees located in the interior.

- $G^c_{i,j} \in \{0,1\}$ be a binary variable indicating whether cell (i,j) is colored with color $c \in \{R,G,B\}$.

P be the set of periphery cell positions:

$$P = \{(0,j),(rows-1,j)\}_{j=0}^{cols-1} \cup \{(i,0),(i,cols-1)\}_{i=1}^{rows-2}$$

- C^c be the total number of tiles of color $c \in \{R,G,B\}$, provided as input.



DIAGONAL CONSTRAINT

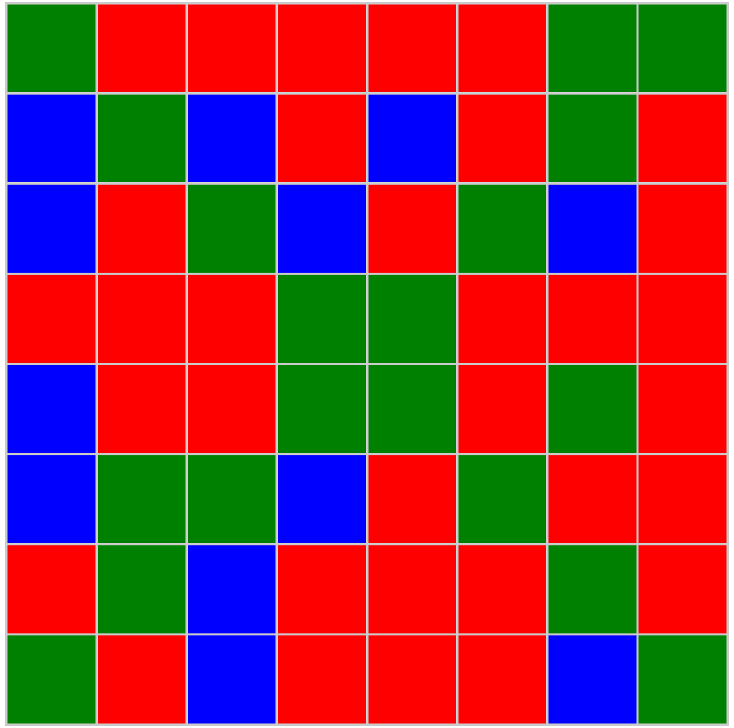
Objective function:

$$\max \sum_{(i,j)\in\mathcal{D}} \sum_{c\in\{R,G,B\}} w_c \cdot G_{i,j}^c$$

The architects want to sit in a way that they can see each other across the room at an angle, keeping their team spread out evenly. How should the office floor be arranged?

Choose the Diagonal Constraint. This algorithm will strategically place architects along the diagonals of the grid, ensuring even spacing and visibility across the room.

- $G_{i,j}^c \in \{0,1\}$ be a binary variable indicating whether cell (i,j) is colored with color $c \in \{R,G,B\}$.
- Let $\mathcal{D} = \{(i,i)\} \cup \{(i, n-1-i)\}$ be the set of all **diagonal positions**, where $n = \text{dimension}$



ADJACENCY CONSTRAINT

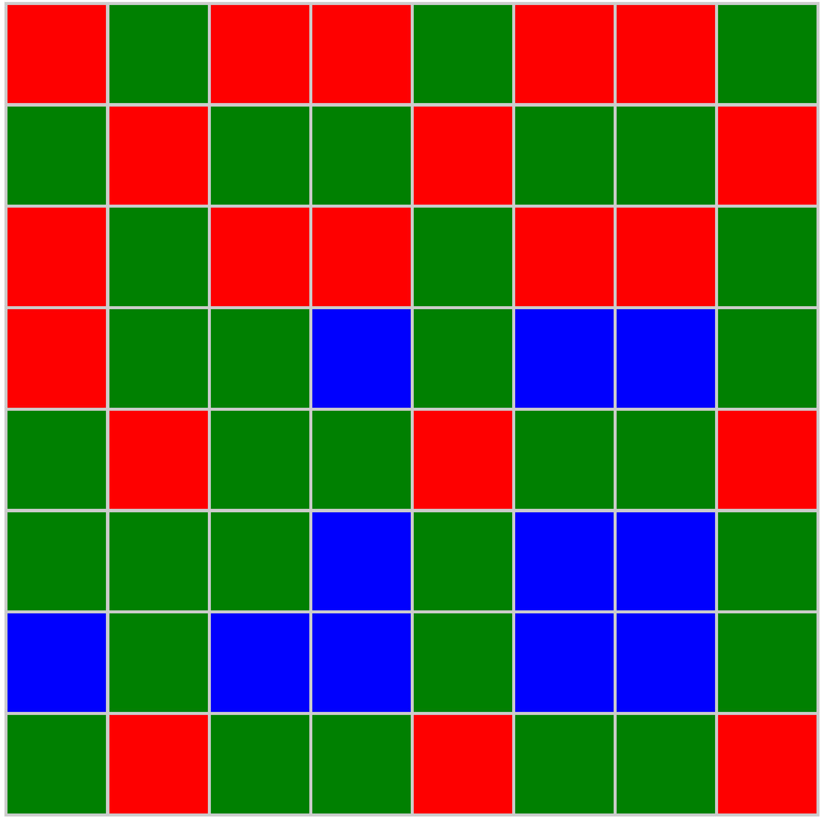
Objective function

$$\text{Maximize} \quad \sum_{i,j} \left(x[i,j]^R \cdot x[i,j+1]^G + x[i,j]^R \cdot x[i,j-1]^G + x[i,j]^R \cdot x[i-1,j]^G + x[i,j]^R \cdot x[i+1,j]^G \right)$$

if x[i, j]^R ==1, then: [i, j+1]^G + x[i, j-1]^G + x[i-1, j]^G + x[i+1, j]^G >= 1

The designers need to sit side by side with the developers to share ideas quickly, while other teams can spread out. How should the office be set up?

Choose the Adjacency Constraint. This will ensure designers and developers are placed next to each other while allowing for a more dispersed arrangement of other teams.



BIG COLORED BLOCK

Objective function

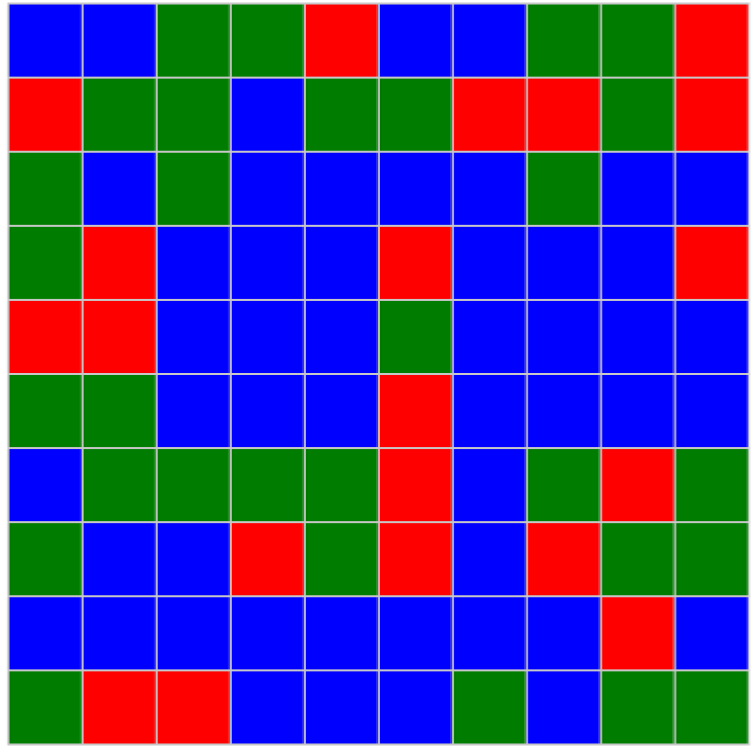
$$\max \sum_{r=1}^{n-s+1} \sum_{c=1}^{m-s+1} y_{r,c}$$

All the developers need to work together in one big group to tackle a major project, with other teams filling in around them. How should the office be laid out?

Choose the "Big Colored Block" constraint. This will allow you to cluster the developers (one color) into a large, central block, while other teams can be placed around them.

Block placement limit

$$\sum_{r=1}^{n-s+1} \sum_{c=1}^{m-s+1} y_{r,c} \leq \min\left(\text{block\_count}, \left\lfloor \frac{\text{color\_budget}}{s^2} \right\rfloor\right)$$



PATTERN CONSTRAINT

Objective function:

$$\max \sum_{r=1}^{\text{row}} \sum_{s=1}^{\text{col}-\ell+1} y_{r,s}$$

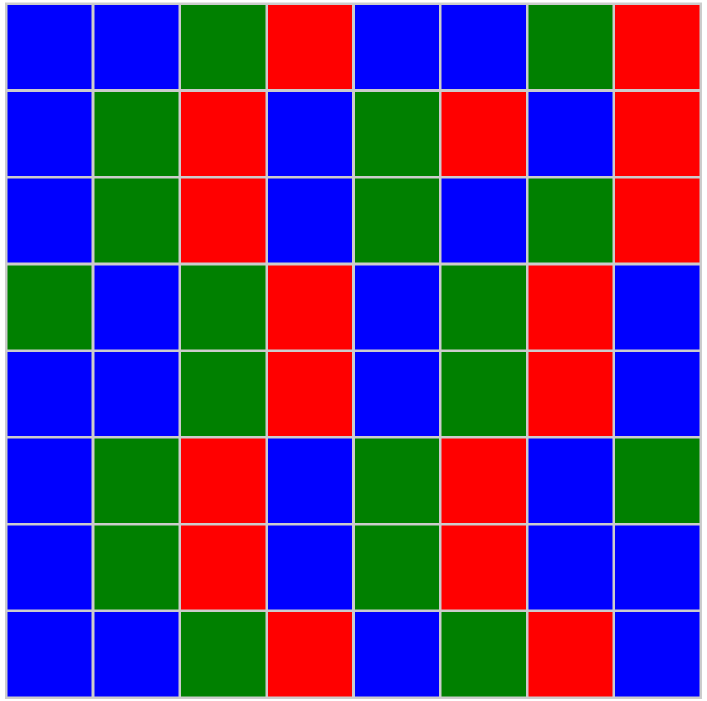
The HR team wants engineers, designers, and architects seated in a repeating sequence of engineers, designers, architects across the floor to balance creativity and focus. They need this arrangement. What's the ideal setup?

Choose the Pattern Constraint. This will allow you to specify the repeating "engineers, designers, architects" sequence across the floor plan.

Pattern enforcement:

If a pattern is placed starting at (r, s), then each cell $(r, s+t-1)$ must match p_t.

$$x^{p_t}_{r,s+t-1} \geq y_{r,s} \quad \forall r, s \in [1, \text{col} - \ell + 1], \ \forall t \in [1, \ell]$$



NO TWO ADJACENT TILES CAN HAVE THE SAME COLOR

Energy function (that is negative of Objective function is)

$$\min \sum_{\text{adjacent } (i,j),(k,l)} \left(x_{i,j}^R \cdot x_{k,l}^R + x_{i,j}^G \cdot x_{k,l}^G + x_{i,j}^B \cdot x_{k,l}^B \right)$$

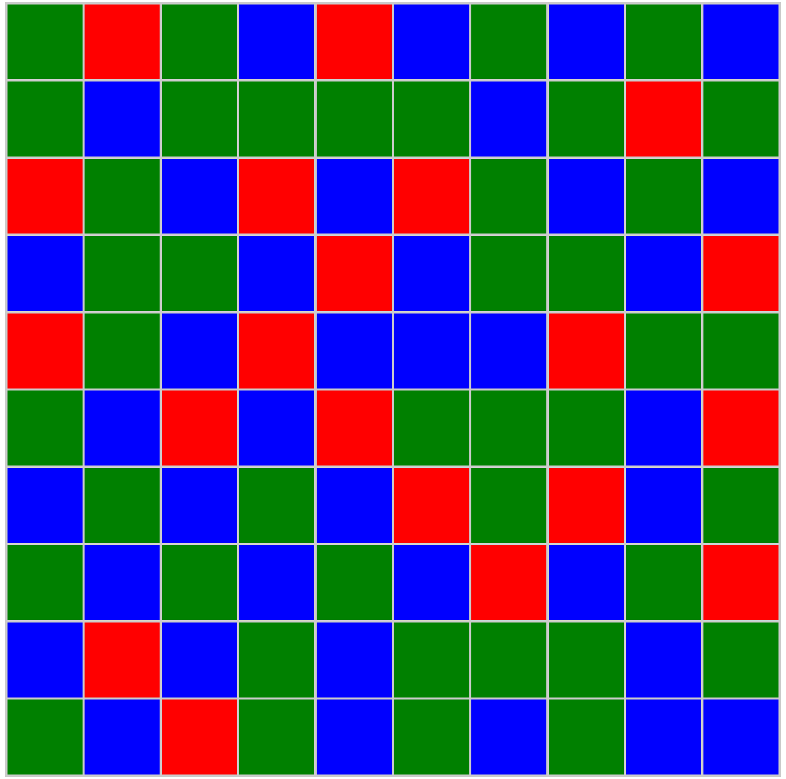
For every pair of adjacent cells $(i, j) \sim (k, l)$:

$$x_{i,j}^R + x_{k,l}^R \leq 1 \qquad x_{i,j}^B + x_{k,l}^B \leq 1$$

$$x_{i,j}^G + x_{k,l}^G \leq 1$$

The company wants a mix of engineers, designers, and architects across the floor, ensuring no two people from the same team sit right next to each other as much as possible. What's the best arrangement?

Choose the "No two adjacent tiles can have the same color" constraint. This will ensure a diverse distribution of team members, preventing clusters of the same occupation.



MULTIPLE CONSTRAINTS

Now we apply multiple constraints at the same time :

- Diagonal & Adjacency Constraint
- Adjacency + Periphery +Diagonal
- Periphery and Diagonal (Priority) Constraint
- Periphery & Big Color Block Constraint
- Periphery & Non-adjacency Constraint

DIAGONAL & ADJACENCY CONSTRAINT

Objective Function:

$$\text{Maximize: } Z = w_d \cdot \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{c \in D} x_{i,j,c} + w_a \cdot \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k \in \{R,G,B\}} \text{adjacency\_satisfaction}(i,j,k)$$

Given a priority list $C_1, C_2, \ldots \in \{R, G, B\}$

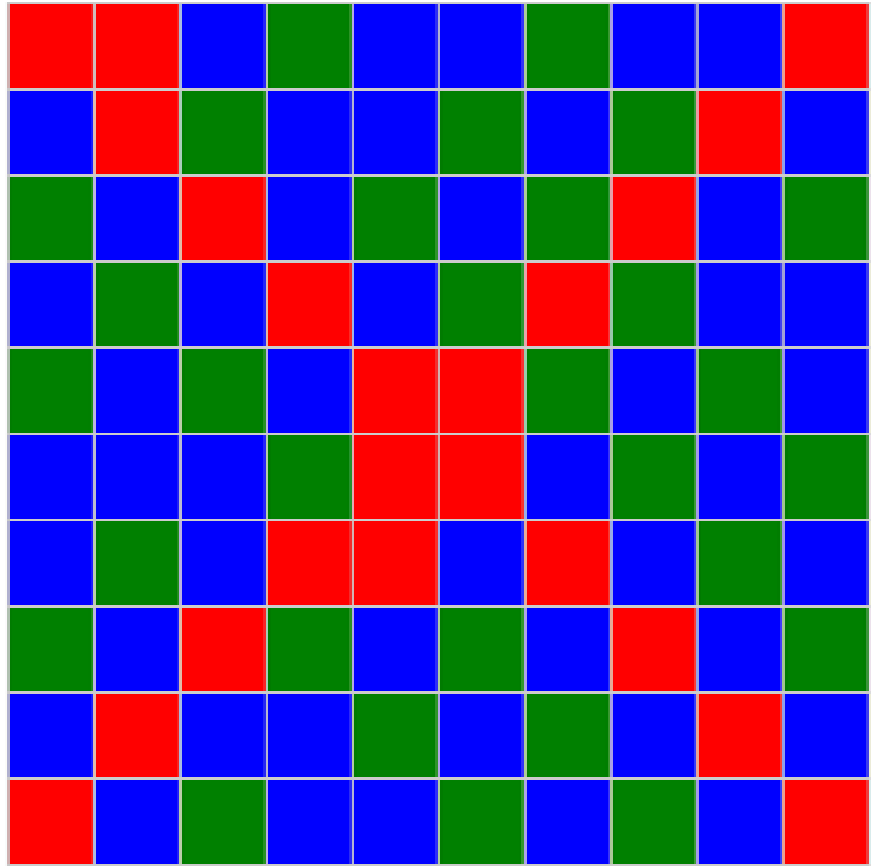
prioritize assigning these colors to the diagonal

$$\sum_{(i,j) \in D} x_{i,j}^{C_k} \geq t_k \quad \text{for some threshold } t_k \text{ for each preferred color } C_k$$

DIAGONAL & ADJACENCY CONSTRAINT

The developers need to sit where they can see their team leads across the room at an angle, while staying close to the designers for collaboration, with architects spread out. They need this arrangement. What's the best layout?

Choose the "Diagonal Constraint" to ensure developers and team leads have angled visibility. Supplement this with the "Adjacency Constraint" to facilitate collaboration between developers and designers.



ADJACENCY + PERIPHERY +DIAGONAL

Boltzmann Objective function

the weights are taken equal by defalut but they can be adjust as per user requirement. if user wants to prioritize between the constraints

$$w_1 \sum_{i,j\in\text{valid}} I(x^{\mathrm{R}}_{i,j}=1)\cdot I(x^{\mathrm{G}}_{i,j+1}+x^{\mathrm{G}}_{i,j-1}+x^{\mathrm{G}}_{i-1,j}+x^{\mathrm{G}}_{i+1,j}\geq 1)+w_2 \sum_{i,j\in\text{valid}} I(x^{\mathrm{G}}_{i,j}=1)\cdot I(x^{\mathrm{R}}_{i,j+1}+x^{\mathrm{R}}_{i,j-1}+x^{\mathrm{R}}_{i-1,j}+x^{\mathrm{R}}_{i+1,j}\geq 1)+w_3 \sum_{(i,j)\in\text{periphery}} I(y^{\mathrm{B}}_{i,j}=1)+w_4 \sum_{(i,j)\in\text{diagonal}} I(z^{\mathrm{G}}_{i,j}=1)$$

1. $x^R_{i,j}$, $x^G_{i,j}$

These are **binary decision variables** representing the color of the cell at position (i,j):

- $x^R_{i,j}=1$: cell (i,j) is assigned color **Red**
- $x^G_{i,j}=1$: cell (i,j) is assigned color **Green**

2. $y^B_{i,j}$

- Binary variable indicating whether the **Blue** tile is on the **periphery** at position (i,j):
    - $y^B_{i,j}=1$ if position $(i,j)\in \text{periphery}$ is assigned a Blue tile.

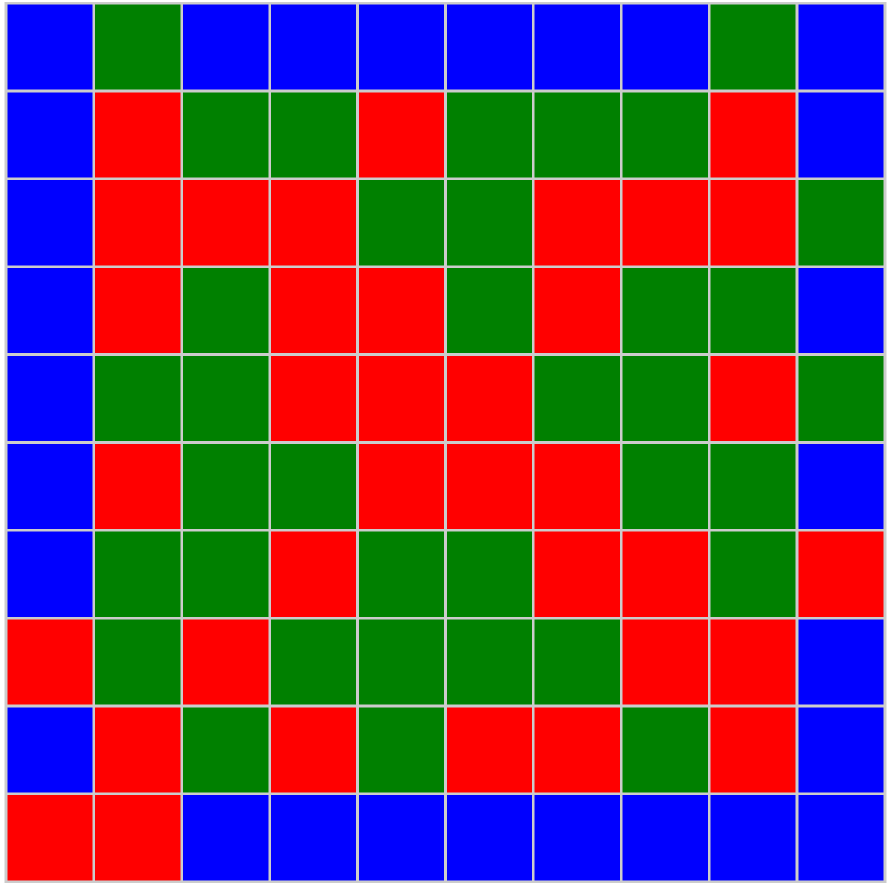
3. $z^G_{i,j}$

- Binary variable indicating whether a **Green** tile is placed on the **diagonal**:
    - $z^G_{i,j}=1$ if position $(i,j)\in \text{diagonal}$ has a Green tile.

ADJACENCY + PERIPHERY +DIAGONAL

The security team should be at the periphery, senior staff diagonally aligned, and designers and engineers should be adjacent. They need this arrangement. How's the office set up?

Choose the "Periphery & Diagonal Constraint (Priority)" algorithm, prioritizing the periphery constraint for security placement and then the diagonal constraint for senior staff. Supplement this with the "Adjacency Constraint" to ensure designers and engineers are placed adjacently.



PERIPHERY & BIG COLOR BLOCK

Objective Function

$$\text{Maximize} \sum_{i,j,c} w_{i,j,c} \cdot x_{i,j,c} + W_b \cdot \sum_{i,j} b_{i,j}$$

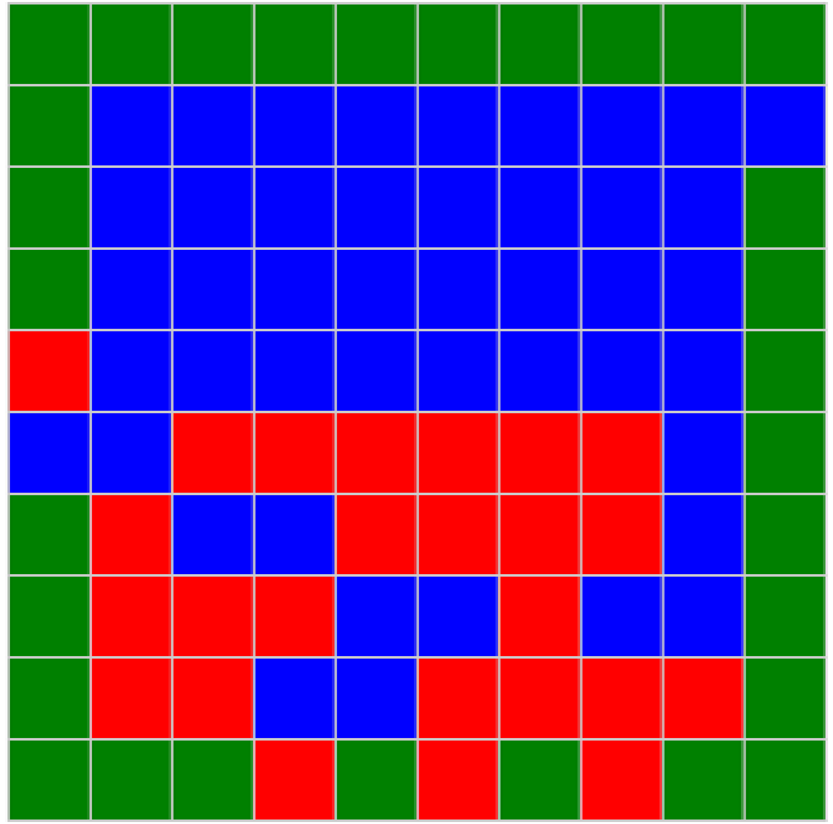
All engineers smoke so they have to sit in a way to not disturb the non-smokers. And all the designers have to sit together as they have to brainstorm the design together. What arrangement is the best?

Choose the "Periphery Constraint" to place smokers (engineers) on the edge, minimizing disruption to non-smokers. Supplement this with the "Big Colored Block Constraint" to cluster designers together for brainstorming.

$$\text{Periphery positions} = \text{col} + \text{col} + (\text{row} - 2) + (\text{row} - 2) = 2 \times \text{col} + 2 \times (\text{row} - 2)$$

$$\boxed{\text{Block Occupancy} = \text{Block Size}^2}$$

$$\boxed{\text{Max Placeable Blocks} = \min\left(\text{Requested Blocks}, \left\lfloor \frac{\text{Available Color Cells}}{\text{Block Occupancy}} \right\rfloor\right)}$$



PERIPHERY & NON-ADJACENCY

Objective Function

$$\text{Maximize} \sum_{i,j,c} w_{i,j,c} \cdot x_{i,j,c} + W_b \cdot \sum_{i,j} b_{i,j}$$

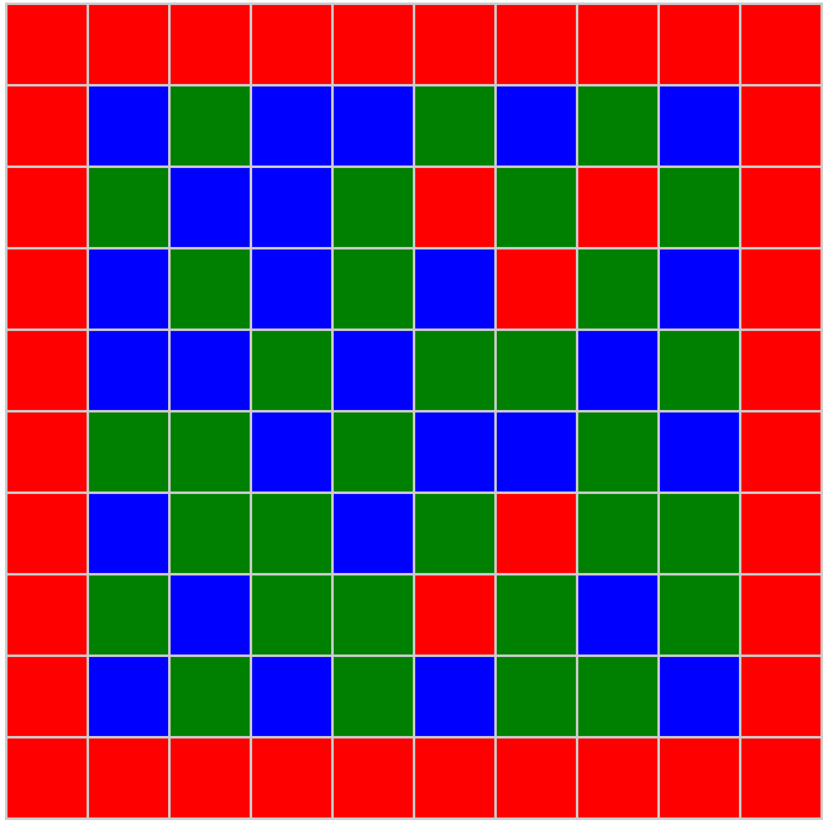
Given that engineers require window-side rooms for smoking, how can we design the workspace layout so that professionals from the same domain — such as architects, designers, and engineers — are not seated adjacent to one another, in order to encourage interdisciplinary interaction and enhance workplace diversity?

Choose the "Periphery Constraint" to place engineers along the perimeter for window access. Then, apply the "No two adjacent tiles can have the same color" constraint to ensure interdisciplinary mixing in the interior.

$$\text{Periphery positions} = \text{col} + \text{col} + (\text{row} - 2) + (\text{row} - 2) = 2 \times \text{col} + 2 \times (\text{row} - 2)$$

$$\boxed{\text{Block Occupancy} = \text{Block Size}^2}$$

$$\boxed{\text{Max Placeable Blocks} = \min\left(\text{Requested Blocks}, \left\lfloor \frac{\text{Available Color Cells}}{\text{Block Occupancy}} \right\rfloor\right)}$$



PERIPHERY AND DIAGONAL

Objective Function

$$w_1 \sum_{i,j} \mathbf{I}(x^R_{i,j} = 1) \cdot \mathbf{I}\left(\sum_{(p,q)\in\mathcal{N}(i,j)} x^G_{p,q} \geq 1\right) + w_2 \sum_{i,j} \mathbf{I}(x^G_{i,j} = 1) \cdot \mathbf{I}\left(\sum_{(p,q)\in\mathcal{N}(i,j)} x^R_{p,q} \geq 1\right) + w_3 \sum_{(i,j)\in\text{periphery}} \mathbf{I}(x^B_{i,j} = 1) + w_4 \sum_{(i,j)\in\text{diagonal}} \mathbf{I}(x^G_{i,j} = 1)$$

Let:

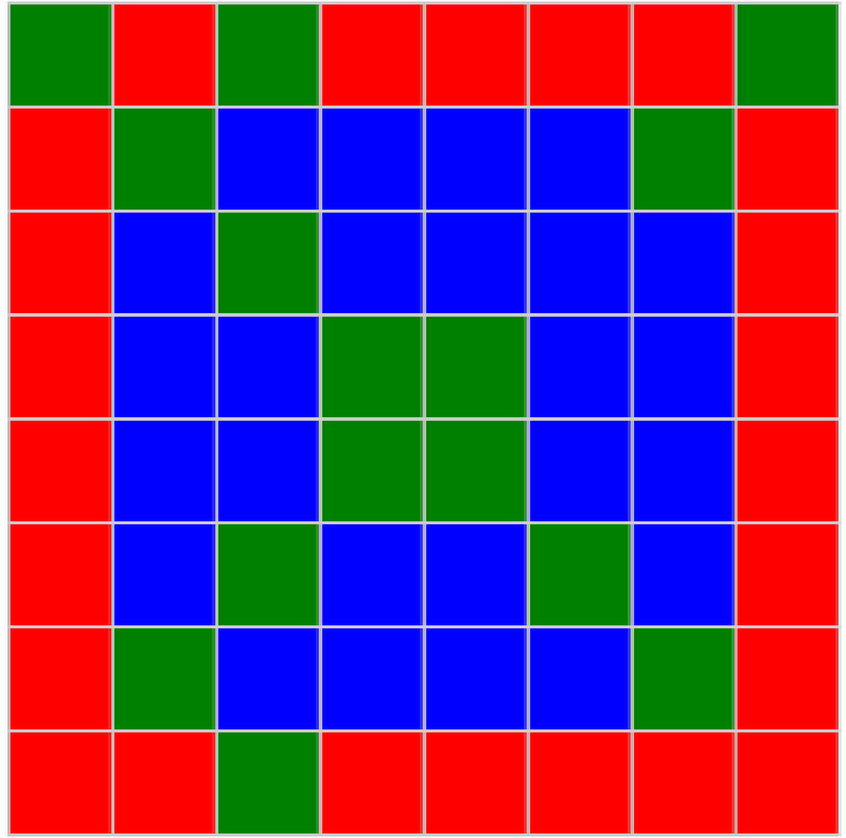
$$\delta_c = \begin{cases} 1 & \text{if } c \text{ is a preferred color for that region} \\ 0 & \text{otherwise} \end{cases}$$

To **encourage** preferred colors (as soft constraints), you **maximize**:

$$\sum_{(i,j)\in P} \sum_{c\in C_P} \delta_c \cdot x^c_{i,j} + \sum_{(i,j)\in D} \sum_{c\in C_D} \delta_c \cdot x^c_{i,j}$$

Engineers and Architects wants seats near the windows and doors and designers want seats so that they see each other across the room at an angle, keeping their team spread out evenly. The designers are given more priority. They need this arrangement. How’s the office laid out?

Choose the "Periphery & Diagonal Constraint (Priority)" algorithm, prioritizing the diagonal constraint for designers and placing engineers and architects along the periphery. This will satisfy the requirements for window/door access and angled designer placement while maintaining even distribution.



CONCLUSION

The application effectively addresses and satisfies a wide range of user-defined constraints commonly encountered in real-life building and workspace design scenarios. These include functional requirements such as room allocation based on user needs, adjacency restrictions to enhance diversity, spatial efficiency, accessibility, natural lighting, and comfort. By integrating these diverse factors into the layout logic, the app ensures that the final design is both practical and user-centric, aligning with real-world expectations and standards.