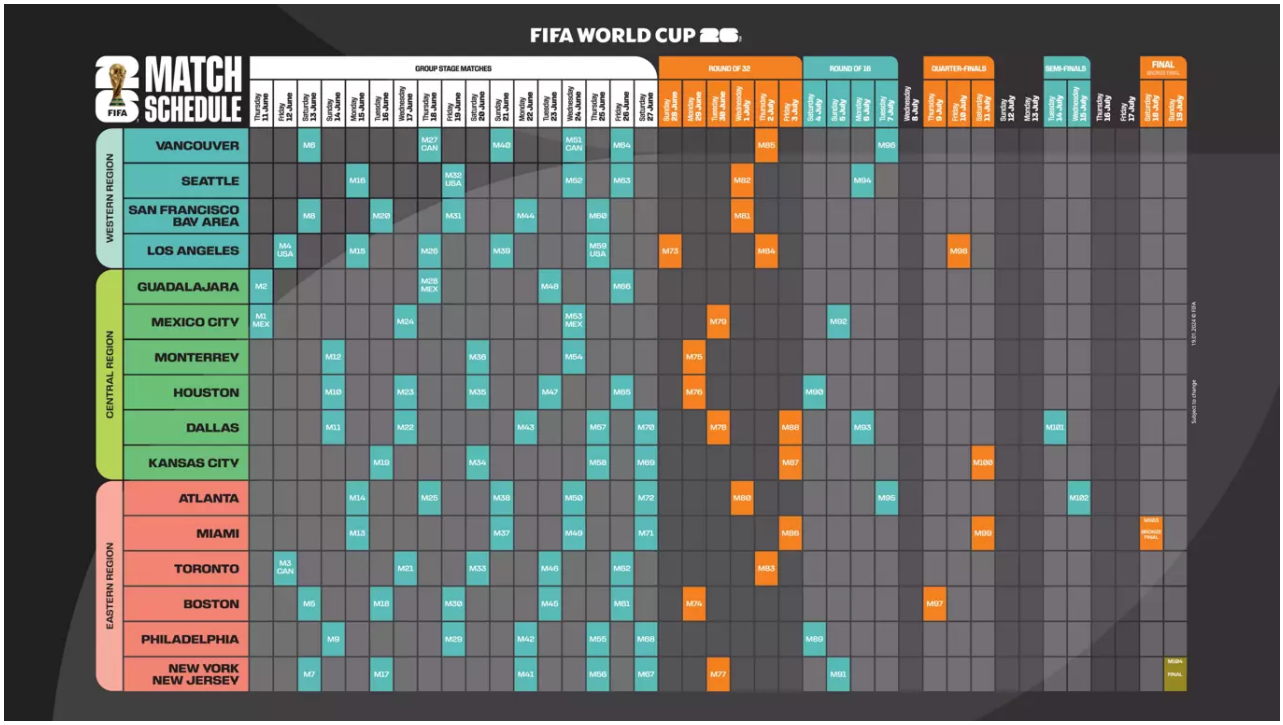
FIFA 2026 World Cup Scheduler:

For this project, we are looking to tackle the game scheduling problem of the FIFA 2026 world cup. Consider the following pre-emptive schedule from the FIFA website:



As can be seen, each game is pre-determined to be played at a certain location within either Canada, the USA, or Mexico. The time and location of the games are locked in, however, the specific teams which will play each game have yet to be scheduled. This is what our project would be looking to determine. Due to the nature of the sport and the popularity of the world cup, it is crucial that the game's schedule is beneficial to both the players, as well as the hosting cities.

Note: For the purpose of this project, we will only be looking to schedule the group stages. This is because in order to schedule the "round of 32", we must know which teams make it past the group stage. Although it is possible to predict the results of the group stage, that is not within the scope of this project, and is also quite indeterministic.

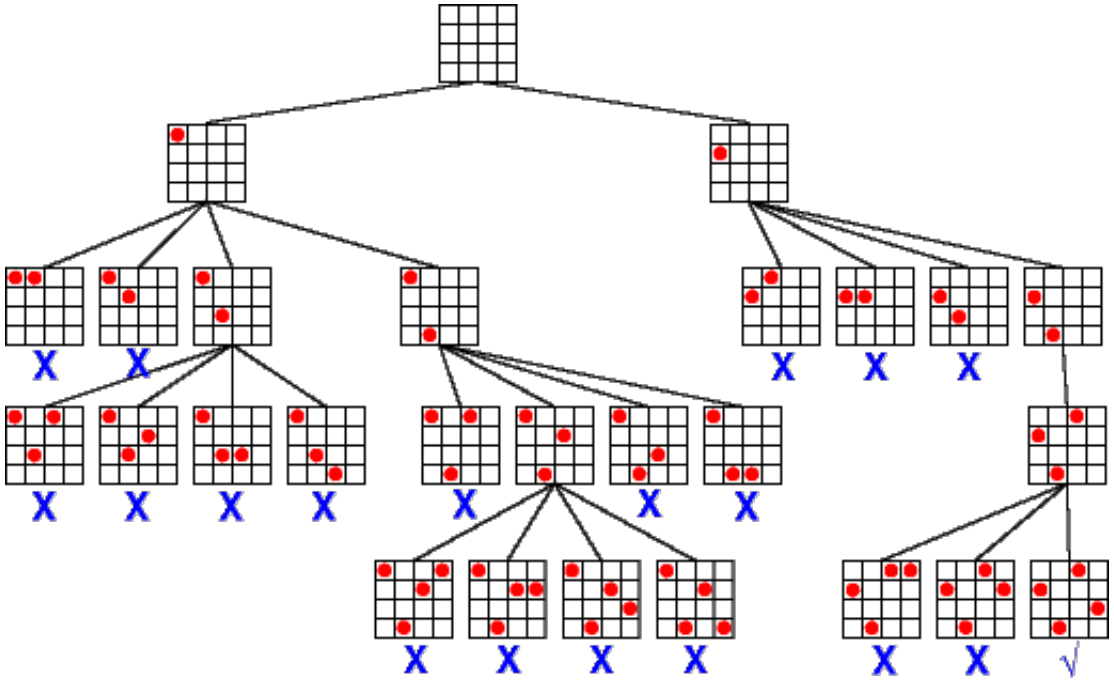
Constraints:

To further specify the problem, we have several constraints that our scheduler must adhere to:

- 48 unique teams
- 12 independent groups (4 teams per group)
- 3 regions (Western has 3 groups, Central has 4 groups, Eastern has 5 groups)
- 3 Games per team (play all other teams within group once)
- 3 Days of rest (at least) between games per team
- Most popular teams play within the bigger stadiums ("most popular" to be defined later on)

Current Approach:

The approach we are going to be taking is similar to that of a Sudoku solver. In particular, we will be using techniques from Constraint Satisfaction and Backtracking to recursively check the feasibility of a particular solution, then subsequently "prune out" infeasible solutions from our solution set. Once we have a set of feasible solutions, we can take the "optimal" solution as defined by our objective function. For example, we can model our set of feasible solutions as a depth-first search tree:



We can assume that the root is the initial state of the schedule (empty). At each layer, we add a pair of two teams to any particular match time slot, and check if that pairing is considered feasible by our constraints. If it is, then we can recursively add a second pair of teams to a new match slot and again check if it is feasible. If it is not feasible, then we can "prune" out that branch such that no further recursions are made with those two teams within that initial slot. This way, we are only left with branches that are considered feasible, and subsequently leaf nodes which contain feasible solutions. Once we have a set of all feasible solutions, we can maximize our objective function over all solutions such that we keep the solution which produces the highest revenue/popularity for FIFA (function will be defined in the next section).

Our data structure will be similar to the above diagram, where each node of the tree will be an nd-array containing the current feasible schedule, and each branch below will be a progression of that schedule. To begin, the root node will have many possible children (since the first match can likely be almost all pairings), however as the tree progresses, the nodes will get pruned out.

This approach is essentially a glorified brute-force search, however, due to the small input space combined with the tight constraints for the teams, we believe a brute force approach won't be much of a computational burden.

Technical Specifics:

Let's define the following variables:

- d_i corresponds to day i of the group stages. Thus, $d_i \in \{d_0, ..., d_D\}$ where we have D days total.
- s_j corresponds to stadium j within the regions. Thus, $s_j \in \{s_0, ..., s_S\}$ where we have S stadiums total.
- t_k is a unique team. Thus, $t_k \in \{t_0, ..., t_T\}$ where we have T teams total.
- g_m is a unique group containing $t_{4m}, t_{4m+1}, t_{4m+2}, t_{4m+3}$ where $m \in \{0, ..., \frac{T}{4}\}$.
- m_n is a unique team matching where $m_n \in \{m_0, ..., m_M\}$ where we have M unique match-ups total.
- $m(t_k, t_l)$ is a mapping function that maps pairs of teams to matches. If a match exists, return m_n, else, return 0. Conversely, we can map $t(m_n)$ that returns the teams t_k, t_l corresponding to the certain match.
- $X(d_i, s_j, t_k, t_l) \in \{0, 1\}$ where 1 if team t_k plays team t_l on day d_i at stadium s_j, else 0.

Additionally, we can define the given constraints in terms of the above variables:

- Restrict schedule to only valid official game days:

$$\text{for all } i, j, k, l; \quad X(d_i, s_j, t_k, t_l) = 0 \quad \text{if not aligned with official FIFA schedule}$$

- Each team plays once per day:

$$\text{for all } d, t; \quad \sum_{j=0}^{S} \sum_{l=0}^{T} X(d, s_j, t_k, t) = 1$$

$$\text{for all } d, t; \quad \sum_{j=0}^{S} \sum_{k=0}^{T} X(d, s_j, t, t_l) = 1$$

- 1 match per day day, per stadium:

$$\text{for all } d, s; \quad \sum_{k=0}^{T} \sum_{l=0}^{T} X(d, s, t_k, t_l) = 1$$

- 3 days of rest at least per team in between games:

$$\begin{aligned} &\text{if} \quad X(d_i, s_j, t_k, t_l) = 1, \quad \text{then} \\ &\quad \text{for all } s \\ &\qquad [X(d_{i+1}, s, t', t) + X(d_{i+2}, s, t', t) + X(d_{i+3}, s, t', t)] = 0 \quad \text{for all } t \text{ and } \quad t' \notin \{t_k, t_l\} \\ &\qquad [X(d_{i+1}, s, t, t') + X(d_{i+2}, s, t, t') + X(d_{i+3}, s, t, t')] = 0 \quad \text{for all } t \text{ and } \quad t' \notin \{t_k, t_l\} - \end{aligned}$$

- Each team can only play another team within it's group (and not itself):

$$\text{if} \quad X(d_i, s, t_k, t_l) = 1, \quad \text{then}, \quad t_k, t_l \in g_m \quad \text{and} \quad t_k \neq t_l$$

Once we have the variables and constraints, we can define the objective function as

$$\text{maximize} \quad \sum_{l=0}^{T} \sum_{k=0}^{T} \sum_{j=0}^{S} \sum_{i=0}^{D} X(d_i, s_j, t_k, t_l) \cdot r(d_i, s_j, t_k, t_l)$$

where $r(d_i, s_j, t_k)$ is a revenue function which returns the estimated revenue generated by team t_k playing at stadium s_j on day d_i. We will determine r through external data such as international popularity rankings of specific teams, stadium capacity numbers, and nationality bias.