```python
from collections import defaultdict
from copy import deepcopy

class Puzzle:
    def __init__(self, initial_state):
        self.initial_state = initial_state
        self.goal_state = [[1, 2, 3], [4, 5, 6], [7, 8, 0]]  # 0 represents the blank space
        self.rows, self.cols = len(initial_state), len(initial_state[0])

    def is_goal(self, state):
        return state == self.goal_state

    def get_blank_position(self, state):
        for r in range(self.rows):
            for c in range(self.cols):
                if state[r][c] == 0:
                    return r, c

    def get_possible_moves(self, state):
        r, c = self.get_blank_position(state)
        moves = []
        directions = [(1, 0), (-1, 0), (0, 1), (0, -1)]  # Down, Up, Right, Left

        for dr, dc in directions:
            new_r, new_c = r + dr, c + dc
            if 0 <= new_r < self.rows and 0 <= new_c < self.cols:
                new_state = deepcopy(state)
                new_state[r][c], new_state[new_r][new_c] = new_state[new_r][new_c], new_state[r][c]
                moves.append(new_state)

        return moves

    def DLS(self, state, depth):
        if self.is_goal(state):
            return True

        if depth == 0:  # Stop if we have reached depth 0
            return False

        for move in self.get_possible_moves(state):
            if self.DLS(move, depth - 1):
                return True

        return False
```

```
    def IDDFS(self, max_depth):
        for depth in range(max_depth + 1):  # Iterate from 0 to max_depth
            if self.DLS(self.initial_state, depth):
                return True
        return False


# Example usage:
initial_state = [[1, 2, 3],
                 [0, 4, 6],
                 [7, 5, 8]]

puzzle = Puzzle(initial_state)
max_depth = 3  # You can adjust this value

if puzzle.IDDFS(max_depth):
    print("Goal state is reachable from the initial state within max depth.")
else:
    print("Goal state is NOT reachable from the initial state within max depth.")
```

```
Goal state is reachable from the initial state within max depth.
```