

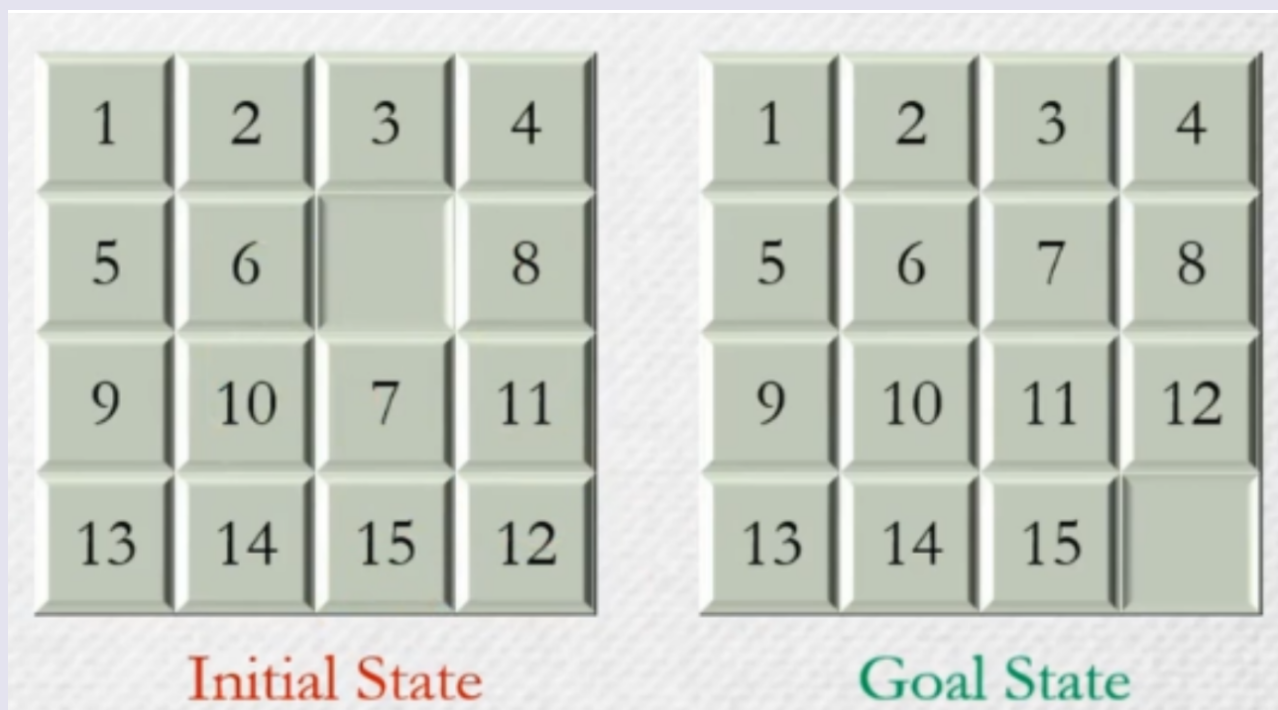
Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 15

**Institute of Computer Technology**  
**B. Tech Computer Science and Engineering**

**Sub: Algorithm Analysis and Design**

**Practical 15**

**Problem** : 15 Puzzle Problem - Given an initial State, reach to the GOAL state using branch and bound algorithm.



**Code:**

```
import YSL_io

GOAL = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, ' ']]
MOVES = [(0, -1), (0, 1), (-1, 0), (1, 0)]
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 15

```
def is_valid(x, y):  
    return 0 ≤ x < 4 and 0 ≤ y < 4  
  
def find_empty(state):  
    for i in range(4):  
        for j in range(4):  
            if state[i][j] == ' ':  
                return i, j  
  
def apply_move(state, move):  
    x, y = find_empty(state)  
    new_x, new_y = x + move[0], y + move[1]  
    if is_valid(new_x, new_y):  
        state[x][y], state[new_x][new_y] = state[new_x][new_y], state[x][y]  
    return True  
  
def is_solved(state):  
    return state == GOAL
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 15

```
def misplaced_tiles(state):  
    count = 0  
    for i in range(4):  
        for j in range(4):  
            if state[i][j] != ' ' and state[i][j] != GOAL[i][j]:  
                count += 1  
    return count  
  
def branch_and_bound(initial_state):  
    queue = [(initial_state, [initial_state])]   
  
    while queue:  
        current_state, path = queue.pop(0)  
  
        if is_solved(current_state):  
            return path  
  
        for move in MOVES:  
            new_state = [row[:] for row in current_state]  
            if apply_move(new_state, move):  
                new_path = path + [new_state]  
                queue.append((new_state, new_path))
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 15

```
queue.sort(key=lambda x: misplaced_tiles(x[0]))

return None


def print_puzzle(puzzle):
    for row in puzzle:
        YSL_io.printCYN(" ".join(f"{cell:2}" for cell in row))

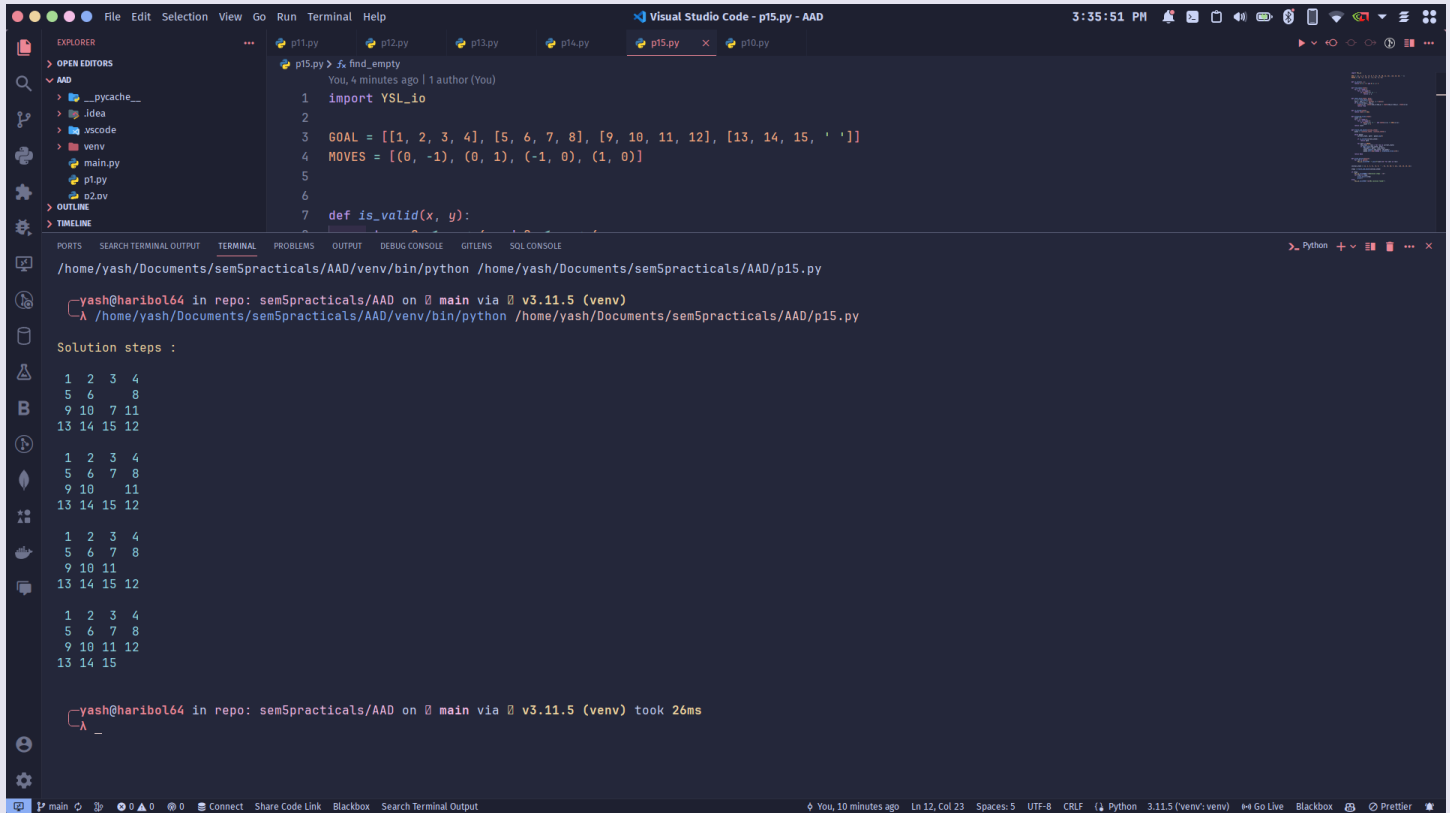

initial_state = [[1, 2, 3, 4], [5, 6, ' ', 8], [9, 10, 7, 11], [13, 14,
15, 12]]

steps = branch_and_bound(initial_state)

if steps:
    YSL_io.printORNG("\nSolution steps : \n")
    for step in steps:
        print_puzzle(step)
        print()
    else:
        YSL_io.printRED("\n\tNo solution found!")
```

Name - Yash Lakhtariya  
Enrollment number - 21162101012  
Branch - CBA      Batch - 51  
AAD Practical 15

## Screenshot:



The screenshot displays the Visual Studio Code interface. The Explorer panel on the left shows a project structure with files like `__pycache__`, `.idea`, `.vscode`, `venv`, `main.py`, `p1.py`, and `p2.py`. The main editor area shows the file `p15.py` with the following code:

```
1 import sys
2
3 GOAL = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12], [13, 14, 15, ' ']]
4 MOVES = [(0, -1), (0, 1), (-1, 0), (1, 0)]
5
6
7 def is_valid(x, y):
```

The terminal window at the bottom shows the command `/home/yash/Documents/sem5practicals/AAD/venv/bin/python /home/yash/Documents/sem5practicals/AAD/p15.py` being executed. The output displays the solution steps for the 15-puzzle problem, showing the initial state and the sequence of moves required to reach the goal state.

```
yash@haribol64 in repo: sem5practicals/AAD on main via v3.11.5 (venv)
/home/yash/Documents/sem5practicals/AAD/venv/bin/python /home/yash/Documents/sem5practicals/AAD/p15.py

Solution steps :

1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 12

1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 12

1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 12

1 2 3 4
5 6 7 8
9 10 11 12
13 14 15

yash@haribol64 in repo: sem5practicals/AAD on main via v3.11.5 (venv) took 26ms
```