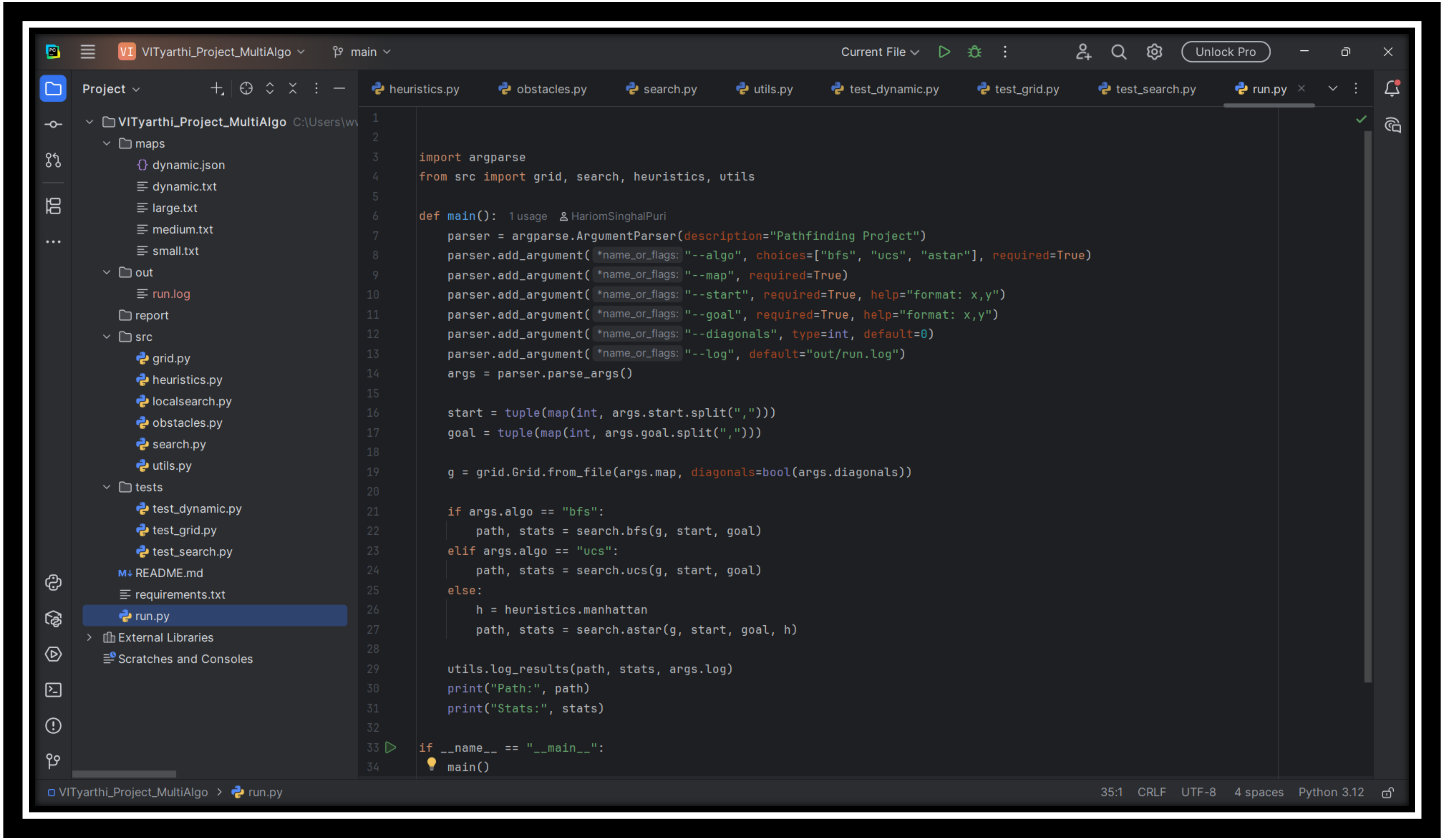
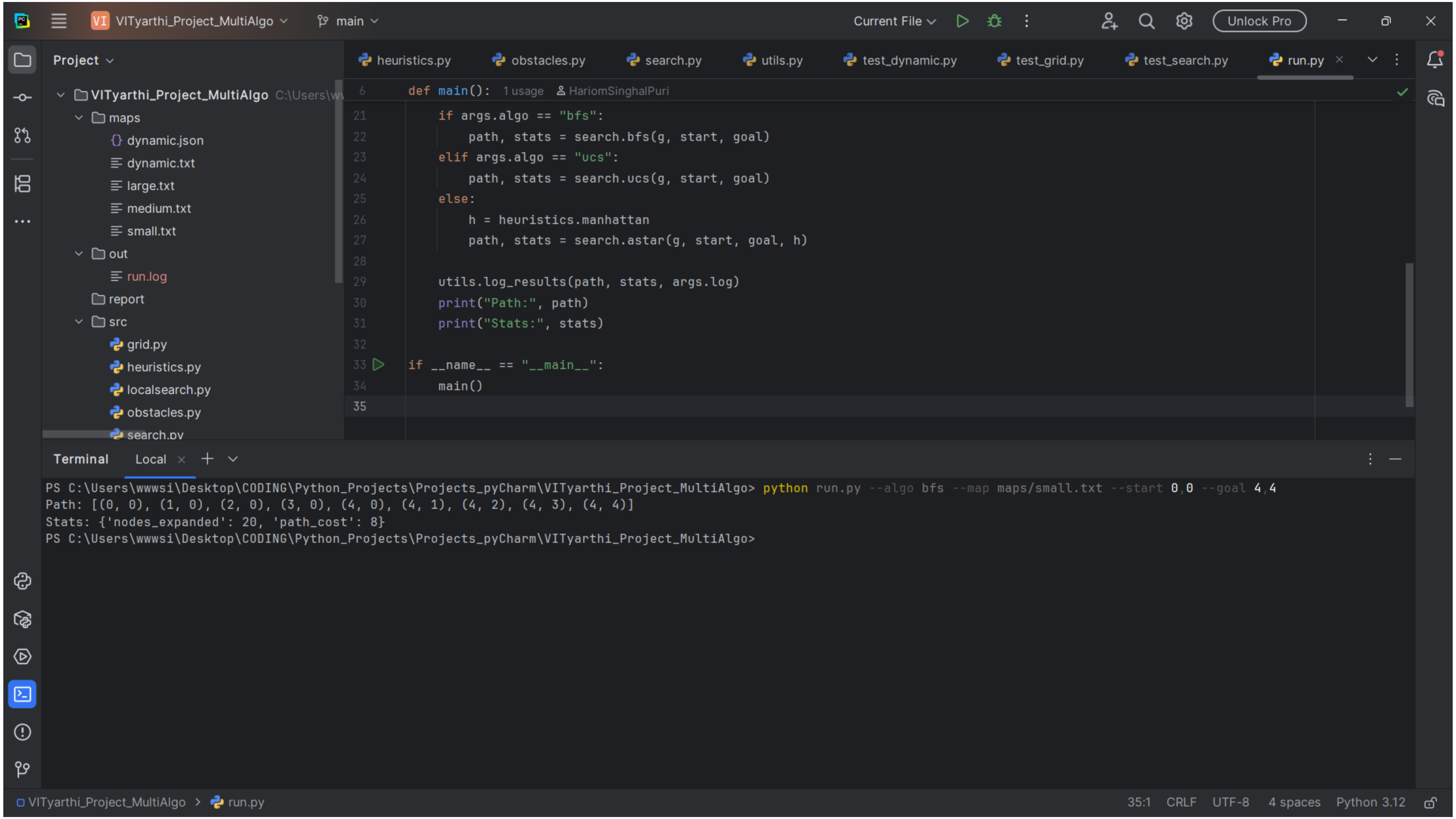
Project Preface



```python


import argparse
from src import grid, search, heuristics, utils

def main():
    parser = argparse.ArgumentParser(description="Pathfinding Project")
    parser.add_argument("--algo", choices=["bfs", "ucs", "astar"], required=True)
    parser.add_argument("--map", required=True)
    parser.add_argument("--start", required=True, help="format: x,y")
    parser.add_argument("--goal", required=True, help="format: x,y")
    parser.add_argument("--diagonals", type=int, default=0)
    parser.add_argument("--log", default="out/run.log")
    args = parser.parse_args()

    start = tuple(map(int, args.start.split(",")))
    goal = tuple(map(int, args.goal.split(",")))

    g = grid.Grid.from_file(args.map, diagonals=bool(args.diagonals))

    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()
```

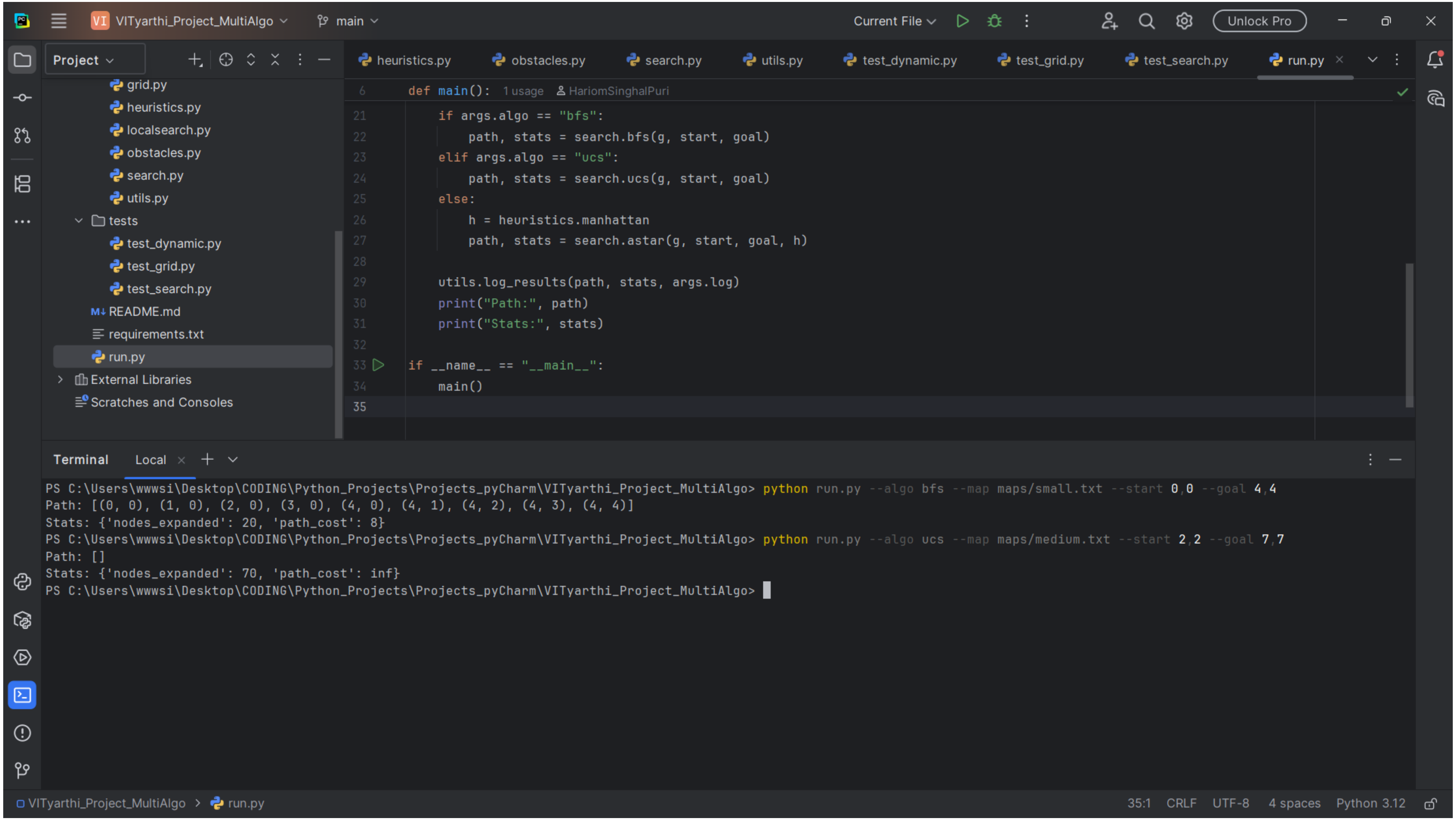
Test 1: BFS on small map



```
def main():
    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()
```

```
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo bfs --map maps/small.txt --start 0,0 --goal 4,4
Path: [(0, 0), (1, 0), (2, 0), (3, 0), (4, 0), (4, 1), (4, 2), (4, 3), (4, 4)]
Stats: {'nodes_expanded': 20, 'path_cost': 8}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo>
```

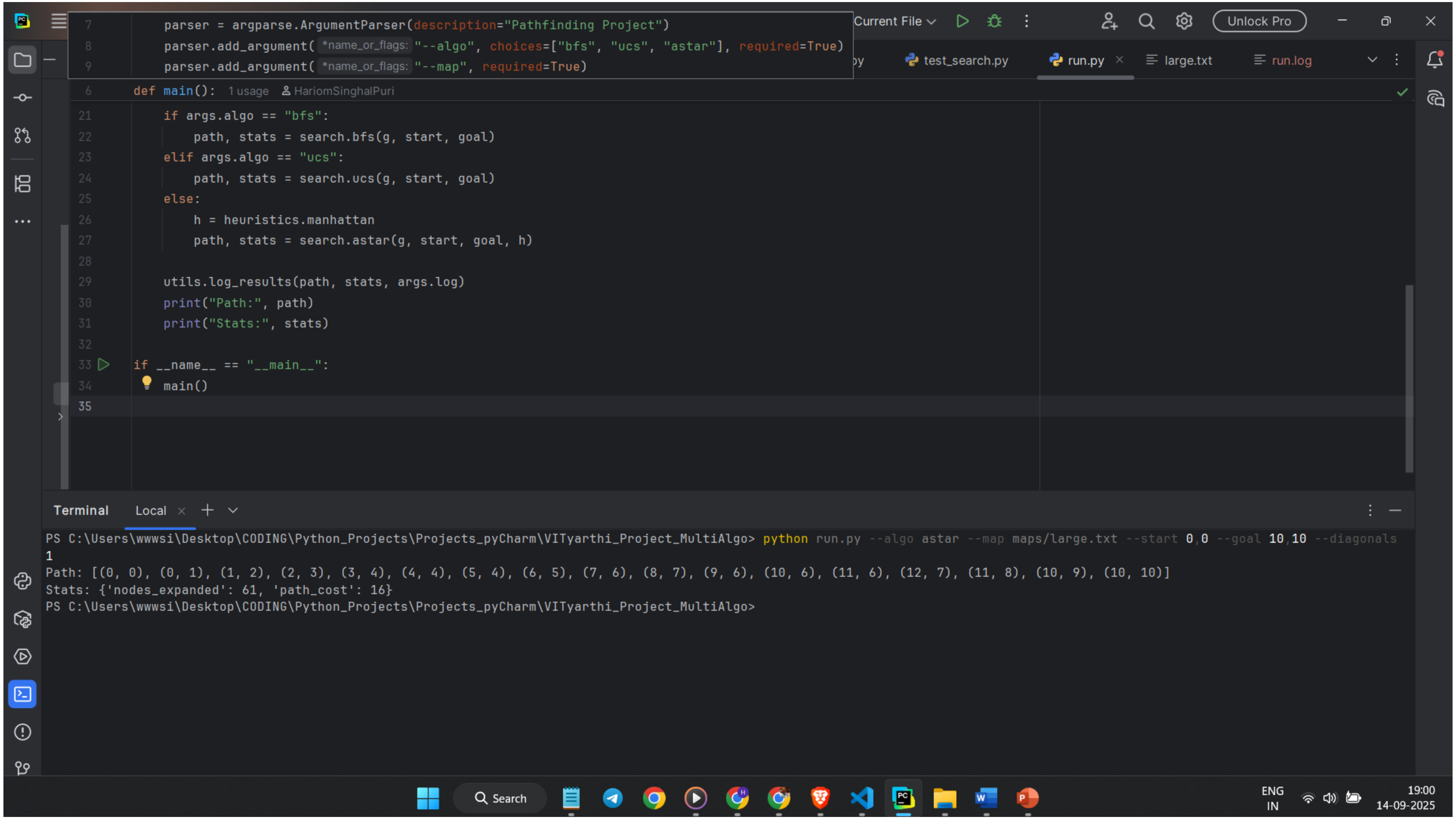
Test 2: UCS on medium map



```
def main():
...
    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()
```

```
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo bfs --map maps/small.txt --start 0,0 --goal 4,4
Path: [(0, 0), (1, 0), (2, 0), (3, 0), (4, 0), (4, 1), (4, 2), (4, 3), (4, 4)]
Stats: {'nodes_expanded': 20, 'path_cost': 8}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo ucs --map maps/medium.txt --start 2,2 --goal 7,7
Path: []
Stats: {'nodes_expanded': 70, 'path_cost': inf}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo>
```

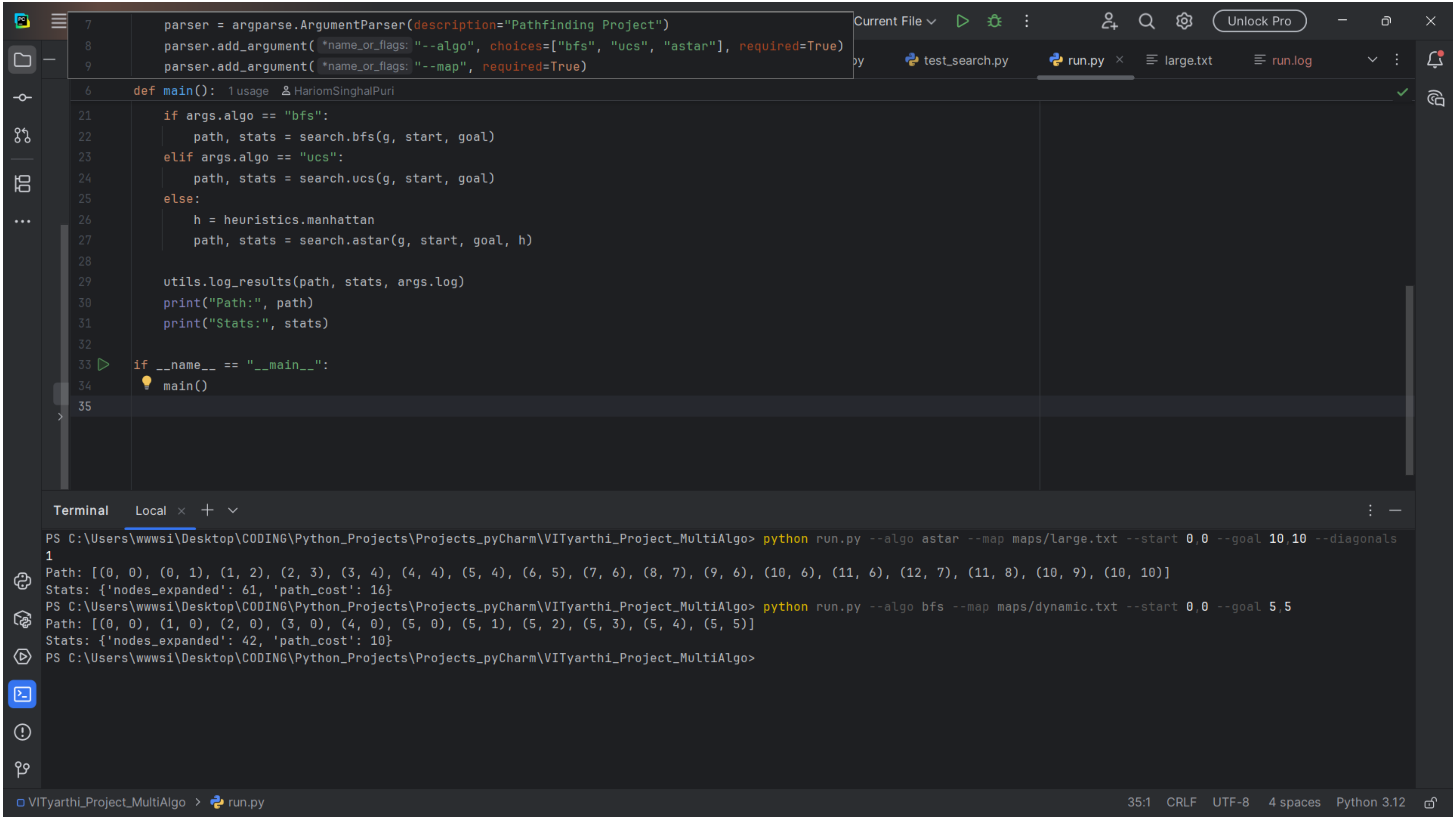
Test 3: A* with large Map allowed



```python
    parser = argparse.ArgumentParser(description="Pathfinding Project")
    parser.add_argument( *name_or_flags: "--algo", choices=["bfs", "ucs", "astar"], required=True)
    parser.add_argument( *name_or_flags: "--map", required=True)
def main():
    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()
```

```
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo astar --map maps/large.txt --start 0,0 --goal 10,10 --diagonals
1
Path: [(0, 0), (0, 1), (1, 2), (2, 3), (3, 4), (4, 4), (5, 4), (6, 5), (7, 6), (8, 7), (9, 6), (10, 6), (11, 6), (12, 7), (11, 8), (10, 9), (10, 10)]
Stats: {'nodes_expanded': 61, 'path_cost': 16}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo>
```

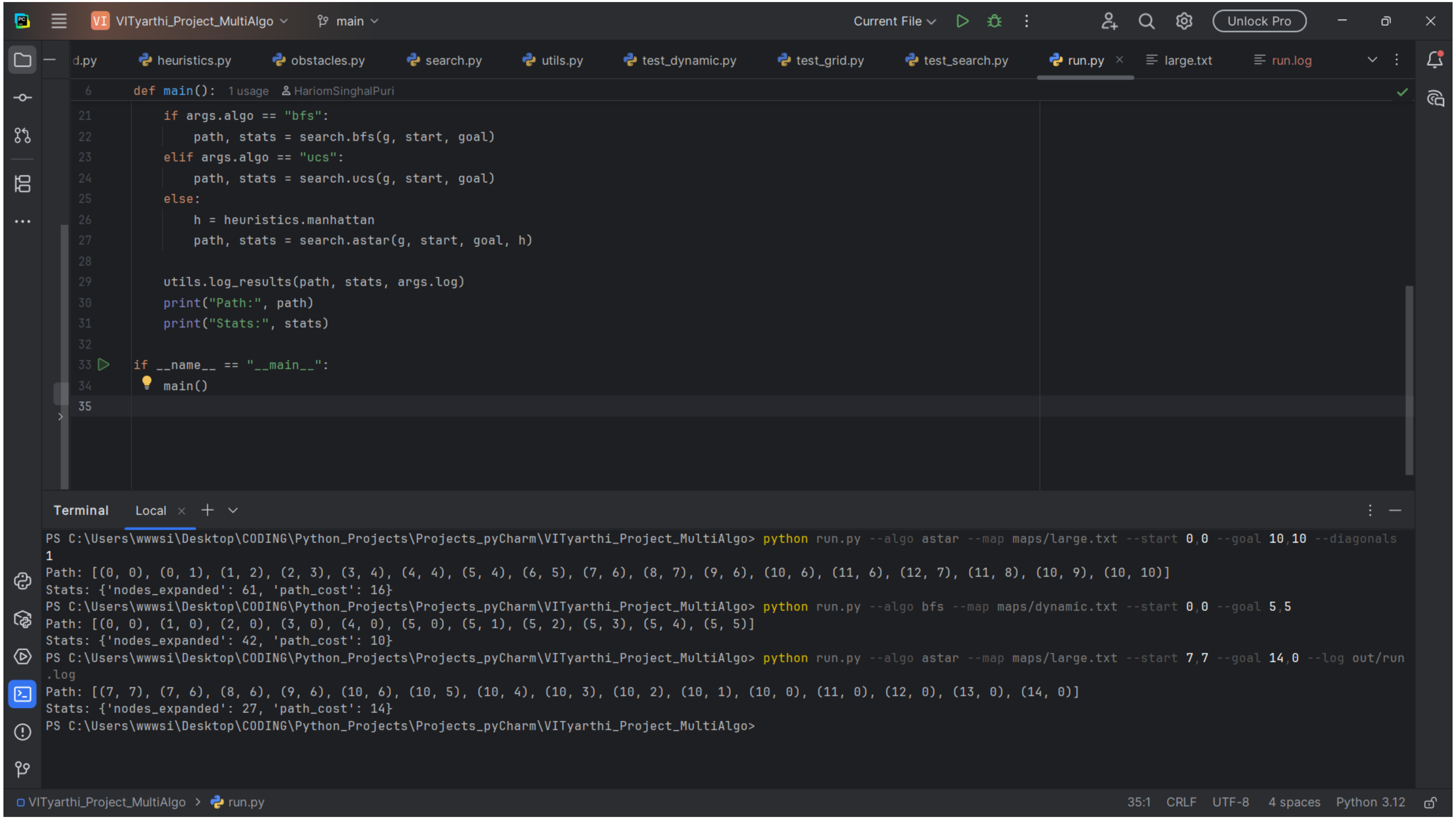
Test 4: BFS on large map with logging



```python
    parser = argparse.ArgumentParser(description="Pathfinding Project")
    parser.add_argument("--algo", choices=["bfs", "ucs", "astar"], required=True)
    parser.add_argument("--map", required=True)
def main():
    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()

```

```
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo astar --map maps/large.txt --start 0,0 --goal 10,10 --diagonals
1
Path: [(0, 0), (0, 1), (1, 2), (2, 3), (3, 4), (4, 4), (5, 4), (6, 5), (7, 6), (8, 7), (9, 6), (10, 6), (11, 6), (12, 7), (11, 8), (10, 9), (10, 10)]
Stats: {'nodes_expanded': 61, 'path_cost': 16}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo bfs --map maps/dynamic.txt --start 0,0 --goal 5,5
Path: [(0, 0), (1, 0), (2, 0), (3, 0), (4, 0), (5, 0), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)]
Stats: {'nodes_expanded': 42, 'path_cost': 10}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo>
```

Test 5: A* on large map, log to custom file



```python
def main():
    if args.algo == "bfs":
        path, stats = search.bfs(g, start, goal)
    elif args.algo == "ucs":
        path, stats = search.ucs(g, start, goal)
    else:
        h = heuristics.manhattan
        path, stats = search.astar(g, start, goal, h)

    utils.log_results(path, stats, args.log)
    print("Path:", path)
    print("Stats:", stats)

if __name__ == "__main__":
    main()

```

```
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo astar --map maps/large.txt --start 0,0 --goal 10,10 --diagonals
1
Path: [(0, 0), (0, 1), (1, 2), (2, 3), (3, 4), (4, 4), (5, 4), (6, 5), (7, 6), (8, 7), (9, 6), (10, 6), (11, 6), (12, 7), (11, 8), (10, 9), (10, 10)]
Stats: {'nodes_expanded': 61, 'path_cost': 16}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo bfs --map maps/dynamic.txt --start 0,0 --goal 5,5
Path: [(0, 0), (1, 0), (2, 0), (3, 0), (4, 0), (5, 0), (5, 1), (5, 2), (5, 3), (5, 4), (5, 5)]
Stats: {'nodes_expanded': 42, 'path_cost': 10}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo> python run.py --algo astar --map maps/large.txt --start 7,7 --goal 14,0 --log out/run
.log
Path: [(7, 7), (7, 6), (8, 6), (9, 6), (10, 6), (10, 5), (10, 4), (10, 3), (10, 2), (10, 1), (10, 0), (11, 0), (12, 0), (13, 0), (14, 0)]
Stats: {'nodes_expanded': 27, 'path_cost': 14}
PS C:\Users\wwwsi\Desktop\CODING\Python_Projects\Projects_pyCharm\VITyarthi_Project_MultiAlgo>
```