

Tents in the Forest

March 8, 2022

1 Find the Tents

Combinatorial Optimization course, FEE CTU in Prague. Created by [Industrial Informatics Department](#).

The problem was taken from <https://www.brainbashers.com/tents.asp> ; there, you can try to solve some examples manually.

1.1 Task

Find all of the hidden tents in the forest grid.

You know that:

- Each tent is attached to one tree (so there are as many tents as there are trees).
- A tent can only be found horizontally or vertically adjacent to a tree.
- Tents are never adjacent to each other, neither vertically, horizontally, nor diagonally.
- A tree might be next to two tents but is only connected to one.

You are also given two vectors indicating how many tents are in each respective row or column of the forest grid.

1.2 Input

You are given a positive integer $n \geq 2$, representing the size of the forest grid (assume it is a square of size $(n \times n)$). You are also given vectors $\mathbf{r} = (r_1, \dots, r_n)$ and $\mathbf{c} = (c_1, \dots, c_n)$ representing the numbers of the tents in the rows and columns of the forest grid. Finally, you are given a list of coordinates of the trees $((x_1, y_1), \dots, (x_k, y_k))$.

```
[23]: # 2x2 - Extra small (for debugging)
n = 3
r = (1, 1, 0)
c = (1, 0, 1)
trees = [(1,1), (3,2)]
```

```
[28]: # 8x8 - Medium
n = 8
r = (3, 1, 1, 2, 0, 2, 0, 3)
c = (2, 1, 2, 2, 1, 1, 2, 1)
trees = [(2, 1), (5, 1), (6, 1),
         (1, 2),
```

```
(3, 3),
(3, 4), (6, 4),
(4, 5), (6, 5),
(8, 7),
(2, 8), (4, 8)]
```

```
[31]: # Weekly special
n = 20
r = (7, 2, 3, 4, 3, 5, 4, 4, 4, 4, 3, 6, 3, 6, 2, 3, 6, 3, 3, 5)
c = (6, 4, 3, 5, 4, 4, 4, 3, 5, 3, 4, 3, 4, 4, 6, 3, 4, 3, 6, 2)
trees = [(3, 1), (4, 1), (8, 1), (13, 1), (15, 1),
          (1, 2), (9, 2), (18, 2), (19, 2),
          (5, 3), (12, 3), (15, 3),
          (2, 4), (4, 4), (9, 4), (17, 4),
          (6, 5), (10, 5), (13, 5), (17, 5), (20, 5),
          (1, 6), (7, 6), (10, 6), (12, 6), (16, 6),
          (20, 7),
          (1, 8), (4, 8), (5, 8), (11, 8), (13, 8), (14, 8), (19, 8),
          (4, 9), (6, 9), (9, 9), (15, 9), (17, 9),
          (8, 10), (17, 10), (19, 10),
          (12, 11),
          (5, 12), (7, 12), (14, 12), (16, 12),
          (1, 13), (2, 13), (6, 13), (19, 13),
          (11, 14), (14, 14), (20, 14),
          (3, 15), (5, 15), (6, 15), (8, 15), (13, 15), (20, 15),
          (2, 16), (3, 16), (10, 16),
          (8, 17), (11, 17), (14, 17), (15, 17),
          (2, 18), (6, 18), (9, 18), (12, 18), (13, 18), (18, 18),
          (2, 19), (7, 19), (15, 19), (17, 19), (20, 19),
          (5, 20), (10, 20)]
```

1.3 Output

You should find the coordinates $(x_i, y_i), i \in \{1, \dots, k\}$, of the individual tents.

1.4 Model

```
[34]: import gurobipy as g

# TODO
```

Visualization

```
[25]: import matplotlib.pyplot as plt
import numpy as np

def visualize(n, trees, tents, r, c):
```

```

grid = [["." for _ in range(n+2)] for _ in range(n+2)]

for t_x, t_y in tents:
    grid[t_y][t_x] = "X"

for t_x, t_y in trees:
    grid[t_y][t_x] = "T"

print(" ", end="")
for c_cur in c:
    print(c_cur, end=" ")
print()

for y in range(1, n+1):
    print(r[y-1], end=" ")
    for x in range(1, n+1):
        print(grid[y][x], end=" ")

    print()

```

[27]: visualize(n, trees, tents, r, c)

```

 1 0 1
1 T . X
1 X . T
0 . . .

```

[30]: visualize(n, trees, tents, r, c)

```

 2 1 2 2 1 1 2 1
3 X T X . T T X .
1 T . . . X . . .
1 . X T . . . . .
2 . . T X . T X .
0 . . . T . T . .
2 . . . X . X . .
0 . . . . . . T
3 X T X T . . . X

```

[33]: visualize(n, trees, tents, r, c)

```

 6 4 3 5 4 4 4 3 5 3 4 3 4 4 6 3 4 3 6 2
7 . X T T X . X T X . . . T X T X . X . .
2 T . . . . . . T . . X . . . . T T X
3 X . . X T . . . . . T . X T . . . .
4 . T . T . X . X T X . . . . X T . . .
3 . X . X . T . . . T . . T . . . T . X T
5 T . . . . . T . X T X T X . X T X . . .

```

```

4 X . . . X . X . . . . . . . . . . X T
4 T . X T T . . . X . T . T T X . X . T .
4 X . . T . T . . T . X . X . T . T . X .
4 . . . X . X . T X . . . . . X . T . T .
3 . . . . . . . . . X T . . . . X . X .
6 . X . X T X T X . . . . X T X T . . . .
3 T T . . . T . . . . X . . . . . X T X
6 X . X . . X . X . . T . X T X . . . . T
2 . . T . T T . T . X . . T . . . . . X T
3 X T T . X . . . . T . . . X . . . . .
6 . . X . . . X T X . T X . T T X . X . .
3 X T . . X T . . T . . T T X . . . T . .
3 . T . . . . T . . X . X . . T . T . X T
5 . X . X T . X . . T . . . . X . X . . .

```

[]: