Ilham Mirzayev
CSCI6511
Project II – Tile Placement
Link to the project - https://github.com/imirzayev/ai-tile-placement

The tile placement project required us to put the given tiles in a way that the final landscape would have desired count of colors. Since the problem is about constraint satisfaction problem, we first need to define what are the variables, domains and constraints. The variables of the problem can be thought of the subdivisions of the general landscape into tile_sizeXtile_size regions. Each region can take one of the tiles available. The final number of bushes must be the one given at input file.

Problem representation in CSP terms

Variables - tile_sizeXtile_size regions. The number of variables are (land_sizeXland_size) / tile_sizeXtile_size.
Domains – {EL-SHAPE, FULL_BLOCK, OUTER_BOUNDARY} if they are available.
Constraints – {The count of tiles (decides whether they are included in the domain or not) and final count of colors.

Actually, the subdivisions of the landscape does not constraint each other in order to benefit the functionalities that CSP techniques provide us. Their constraint is only revealed once any tile is finished or the final count of bushes is violated(or found). Despite I could not use the techniques on relations of arcs with each other, I developed a generic code that could be used when a constraining arcs are given. I used ac-3 algorithm in order to decrease the domains of arcs by using the tile count and bush size as a constraint.

The algorithm I used to solve the problem is recursive backtracking.

I tried several heuristic functions in order to choose which tile to choose next. They were named accordingly in the code.

Heuristic function approaches

heuristic1 – In this function I want to decide which tile to choose next based on the distance from every color to the final color sets. I get the possible cases after placement of each tile, calculate the distance from current colors to the target colors, and considering every color decide on which tile to choose. The tile that cause most of the color to be minimally distanced from the target is chosen.

heuristic2 – Here I choose the next tile based on the sum of each color's distance from its target count. The minimum count is chosen since it causes the state to approach the final state most.

simple_lcv – I this function I want to use the tile which is least constraining. Generally, the el shape tile can be considered as least constraining since it covers the less space. After the outer block, and after full block will be used here.

lcv - The previous function has some shortages since it does not take into account that, there are cells that do not have colors in them. So the previous approach was not completely correct. Here I consider the existence of empty cells, and realize the same approach by counting non zero cells after placement of possible tiles.

I have developed the mentioned functions and used them in my code. Unfortunately, none of them were successful on finding the results. I still have not figured out their shortcomings on doing their tasks.

In my final version of code, I use the constraint satisfactions of tile counts and final bush count at each step and backtrack if I do not find the result.

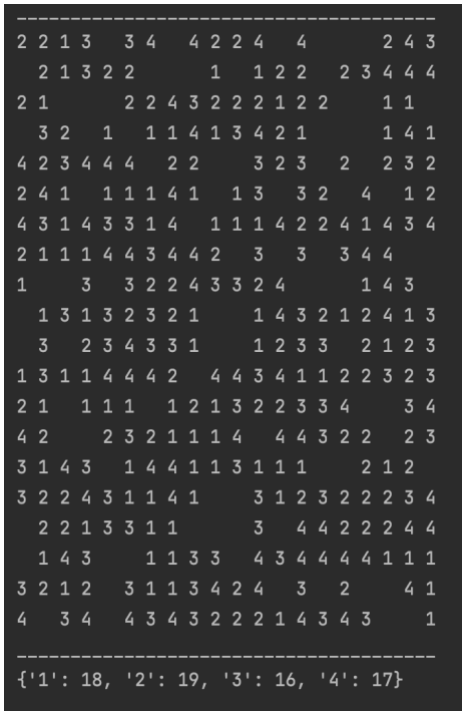
Unit Tests

Unit tests have been implemented in order to guarantee the correctness of program at different stages. Test_inputparser.py file contains tests to ensure that the input files are read and parsed correctly. Test_landscape.py file contains tests about operations on landscape. It makes sure that, different types of tile placement on the landscape and the counting of colors is done correctly. Finally, test_algorithm.py unit test checks the color count of the found result with the color count of its target.

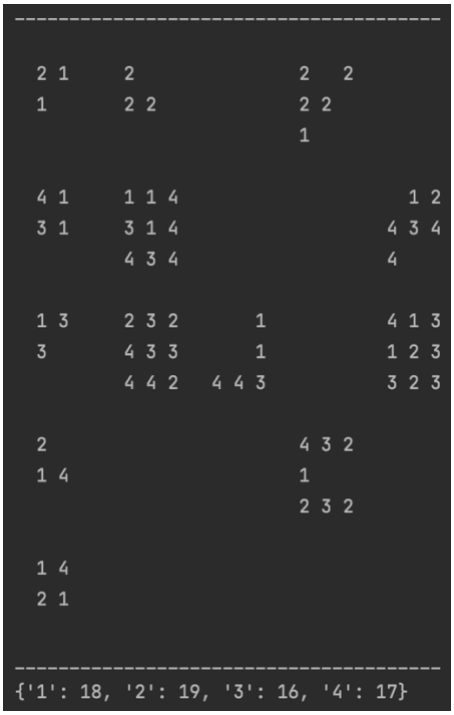
The program outputs the initial version of landscape, its target color counts, the found updated landscape, the current color counts of found landscape and the order in which the tiles are placed on the landscape.

Below I have showed some of outputs by the program.
How to run? - Use main.py to include the input file name and run the code.

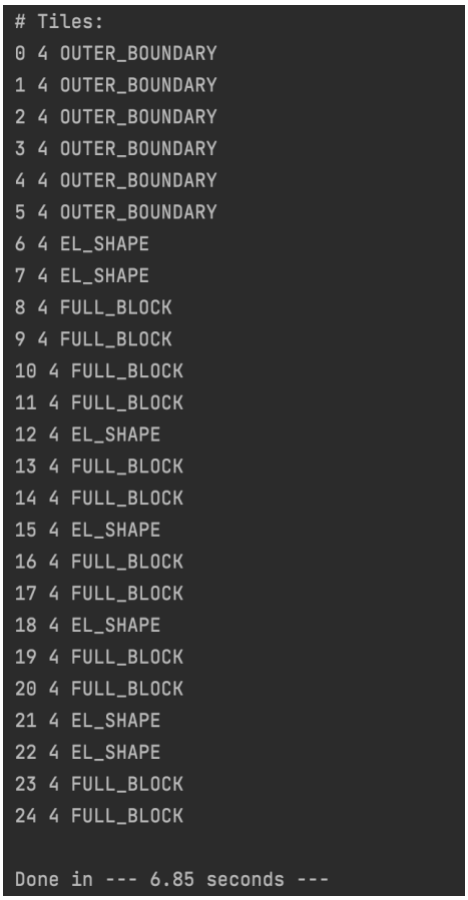
tilesproblem_1326658913086500.txt



```
------------------------------------
2 2 1 3   3 4   4 2 2 4   4       2 4 3
  2 1 3 2 2       1   1 2 2   2 3 4 4 4
2 1       2 2 4 3 2 2 2 1 2 2     1 1
  3 2   1   1 1 4 1 3 4 2 1       1 4 1
4 2 3 4 4 4   2 2     3 2 3   2   2 3 2
2 4 1   1 1 1 4 1   1 3   3 2   4   1 2
4 3 1 4 3 3 1 4   1 1 1 4 2 2 4 1 4 3 4
2 1 1 1 4 4 3 4 4 2   3   3   3 4 4
1     3   3 2 2 4 3 3 2 4       1 4 3
  1 3 1 3 2 3 2 1     1 4 3 2 1 2 4 1 3
  3   2 3 4 3 3 1     1 2 3 3   2 1 2 3
1 3 1 1 4 4 4 2   4 4 3 4 1 1 2 2 3 2 3
2 1   1 1 1   1 2 1 3 2 2 3 3 4     3 4
4 2     2 3 2 1 1 1 4   4 4 3 2 2   2 3
3 1 4 3   1 4 4 1 1 3 1 1 1     2 1 2
3 2 2 4 3 1 1 4 1     3 1 2 3 2 2 2 3 4
  2 2 1 3 3 1 1       3   4 4 2 2 2 4 4
  1 4 3     1 1 3 3   4 3 4 4 4 4 1 1 1
3 2 1 2   3 1 1 3 4 2 4   3   2     4 1
4   3 4   4 3 4 3 2 2 2 1 4 3 4 3     1
------------------------------------
{'1': 18, '2': 19, '3': 16, '4': 17}
```



```
------------------------------------


  2 1     2             2   2
  1       2 2           2 2
                        1


  4 1     1 1 4                   1 2
  3 1     3 1 4                 4 3 4
          4 3 4                 4


  1 3     2 3 2       1         4 1 3
  3       4 3 3       1         1 2 3
          4 4 2   4 4 3         3 2 3


  2                     4 3 2
  1 4                   1
                        2 3 2


  1 4
  2 1


------------------------------------
{'1': 18, '2': 19, '3': 16, '4': 17}
```



```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 OUTER_BOUNDARY
3 4 OUTER_BOUNDARY
4 4 OUTER_BOUNDARY
5 4 OUTER_BOUNDARY
6 4 EL_SHAPE
7 4 EL_SHAPE
8 4 FULL_BLOCK
9 4 FULL_BLOCK
10 4 FULL_BLOCK
11 4 FULL_BLOCK
12 4 EL_SHAPE
13 4 FULL_BLOCK
14 4 FULL_BLOCK
15 4 EL_SHAPE
16 4 FULL_BLOCK
17 4 FULL_BLOCK
18 4 EL_SHAPE
19 4 FULL_BLOCK
20 4 FULL_BLOCK
21 4 EL_SHAPE
22 4 EL_SHAPE
23 4 FULL_BLOCK
24 4 FULL_BLOCK

Done in --- 6.85 seconds ---
```

tilesproblem_1326658918378200.txt

```
----------------------------------------
4 4 4 2   1 4 4 3   3       4 2 4 3 4
1 3 2 1 2 4 4   1 3 4 2 2 4 3 1 2 2 1 2
4 3   1 4 1   3 3 3 2     4 2 4     4 2
1 1   1 4 3 4 4   2 4 1 2   4 2 2 3 4 4
2     2 4 2 4 4 1   4 1 3   3 2       4
  3 1 1 4 4 1 3   4   3   4 1 4 1 1 3 3
3 4 4 1 2         2 1 1 3 1 1 1 4 1   2
  4 3 4   3     1   4 4   4   4 4 4
1 3 3 2 1 2 3   4 3 2 4   2 3 4 4 4 4 4
2 2   2 1 3 1 3 4 1 1   2 2 3   2 2 1
  2 1 2 3 1   3 4 4     2 3 3 2 1 2 1 2
2 2 1 1 3     4   1 1 2   2 1 1   1 2 3
2 2   1   3 4 1 2 3 4     2 1 4 4 1 1 1
4 1 1 3 4 3   4 1   4   4 2 3 2 2 2
  4 4   2 4 1 2 1     2   3 1 2 4 4 3 4
3 1   3 3 2   3 2 1 3 3 4 4   4 2 4 4 3
2 4 1 4       3   4 2 3 4 2 2 1 1 3 4
2 4 4   3   2 3 3   1 3   4 1 3 1 4 4 4
1 4 1 4   1   4   1 3 3 3 3 1 2 1 1   1
4     4 1   1 3 1 4 2 4     2 2 3 2   2
----------------------------------------
{'1': 27, '2': 22, '3': 15, '4': 35}
```

```
----------------------------------------

  3 2     4 4             4 3 1   2 1 2
  3       1               4 2 4     4 2
                            4 2   3 4 4

  3 1     4 1 3   4   3   4 1 4
  4 4             2 1 1   1 1 1
          3         4 4   4   4

  2               1 1             2 1
  2 1             4               2 1 2
                  1 1 2           1 2 3

  1 1     3   4                   2
  4 4     4 1 2                   4 3 4
          2   3                   4 4 3

  4 4                     4 1 3   4 4 4
  4 1                     3 1 2   1   1
                            2 2   2   2
----------------------------------------
{'1': 27, '2': 22, '3': 15, '4': 35}
```

```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 OUTER_BOUNDARY
3 4 OUTER_BOUNDARY
4 4 OUTER_BOUNDARY
5 4 OUTER_BOUNDARY
6 4 EL_SHAPE
7 4 FULL_BLOCK
8 4 EL_SHAPE
9 4 FULL_BLOCK
10 4 FULL_BLOCK
11 4 EL_SHAPE
12 4 EL_SHAPE
13 4 FULL_BLOCK
14 4 FULL_BLOCK
15 4 EL_SHAPE
16 4 EL_SHAPE
17 4 FULL_BLOCK
18 4 FULL_BLOCK
19 4 EL_SHAPE
20 4 EL_SHAPE
21 4 FULL_BLOCK
22 4 EL_SHAPE
23 4 EL_SHAPE
24 4 EL_SHAPE

Done in --- 27.51 seconds ---
```

tilesproblem_1326658930331900.txt

```
----------------------------------------
    4 2 3 4 2 2 3 1 2 4   1 2 3 4   4
4 1   1 3 1 2 2 2 1     4 4 4   1 1 1 3
1 3   3   1 1 2   3 2   3 4 4     3 2
1   2 1 3 3 3 4 4 1   2 3 3 2 3 2 3 2 4
2 4 3 2 3 4   1   4 1 3 2 4 2 3   3 4 1
  1 4 2 1 3 3 3 1 2 1 1 4 3 4     1 2 3
2 1 1 3 4 2 4   3 2 4 4 4   2   1   1 1
  4 1 1 4 1 2 3 3   3 1 4 4 1 4     1 2
4 2 2 4   3 2 1 2 4 1 3 3 4 2 4 2 2 2 1
  4 4   4   4   3     2 1 2   2 1 2 3
4 3   3 3 1 3 4 4   3 3   1 2 4   3 3
2 1 3 3 2 4   3 4 4   4 3 4 3   1   4 2
2 1 1 3 3 1 1   3 4   2   3 3 1 2 2 2
3 1 2 3 3 3 2 1 2 4 1 1 4 1   4 4 2 2 1
3 2 4 1 2 3   1   2   4 1 3   3 1 2 3 4
2 3 1 1   1 2   2   4 2 2 4 4   4 3
  1 1 1 3 2 3 4 1 3 2 4 2 3 1 4 1 1 4
4   2 1   1 3 2   2   1   4 4 1   4 4 4
3 3 1 1 3 2   2 2   2 2 2 2 1 4 4 3 3 2
3 2 2 4 1 3 3 2 3 3     4     3 1 3 2 2
----------------------------------------
{'1': 20, '2': 25, '3': 25, '4': 29}
```

```
----------------------------------------

 1       1 2             4 4
 3       1 1             4 4
                         3 2 3

 1 4     3 3             3 4
 1 1     2 4               2
                         4 1 4

 4 4       4         2   2   2
 3       1 3 4     3 3   1 2 4
         4   3   4   4   4 3

 1 2     3 2 1   4 1 1   1   4   2 2 1
 2 4     3   1   2   4   3   3   2 3 4
         1 2       4 2   4 4     3

   2     1 3 2                   4 4 4
 3 1     2   2                   3 3 2
         3 3 2                   3 2 2
----------------------------------------
{'1': 20, '2': 25, '3': 25, '4': 29}
```

```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 OUTER_BOUNDARY
3 4 OUTER_BOUNDARY
4 4 OUTER_BOUNDARY
5 4 OUTER_BOUNDARY
6 4 OUTER_BOUNDARY
7 4 EL_SHAPE
8 4 EL_SHAPE
9 4 EL_SHAPE
10 4 FULL_BLOCK
11 4 FULL_BLOCK
12 4 EL_SHAPE
13 4 EL_SHAPE
14 4 FULL_BLOCK
15 4 EL_SHAPE
16 4 EL_SHAPE
17 4 EL_SHAPE
18 4 EL_SHAPE
19 4 FULL_BLOCK
20 4 FULL_BLOCK
21 4 FULL_BLOCK
22 4 FULL_BLOCK
23 4 EL_SHAPE
24 4 EL_SHAPE

Done in --- 3.71 seconds ---
```

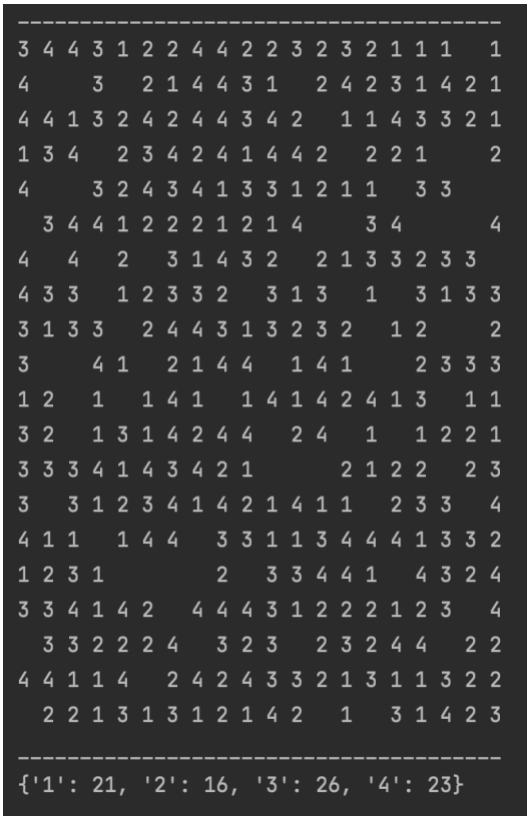
tilesproblem_1326658926570700.txt

```
 ------------------------------------
 4 4 2 4 4 2 4   4 1   3   1 3 3 2 2 1 3
 3     2 4   2   3 4 4 1 4 1 1 3 4 1 4
 1 1   3 2   3   4 4 2   3 2 1   3 2 4 2
 1 4 1 4   4   1 2 3 1 4     3     1 4
   1 2 3 3 4   4 2 1 1 2 4   3 4 2 4 2 4
 3   2 3 3   4 4 4 3 1 2 3 4 1 1 4   3 2
   1 1   1 2 3 2 3 3 4 1 4 3 3 3 2   3 1
   2 2 1 3 3 1 3 1     1 3 3       4 2
   2 3 2 1   3 1 4 2 3 3 3   2 2 3 1 3 2
 1   1 1     4 1 1   3 4   1 3 3 3 3 4 1
 4 2 4 1   4   4   1 1     4 1 4 4 3 4
 4 4 1 1 2     3 2 4 1 2 2 3 2   4   3 1
 2 1 3     4   1 4 1     2   3 2 3 4 4 3
 3 3 2     2 2 3   1 4 4   4 1 1 3 2 1 2
 1 3 4 3   3   2 4 2 2 2 3 1   2 1 3
     3   3 3 2 1 3 1 1 4 1   4 4 1 1 3 3
 3 3 3   4 3 4 3 4 4 1 2 3 3 1   1 3 3 3
 4 1 1 3 3 2   4 2 2 3 3 2 2 1 2 4 1 1 2
   3 3 2 2 2   2 4 2 2 1   3     4 2 3 3
   1   3   4 1 4   4   4 1 1   4 3 4 2 2
 ------------------------------------
 {'1': 23, '2': 18, '3': 20, '4': 18}
```

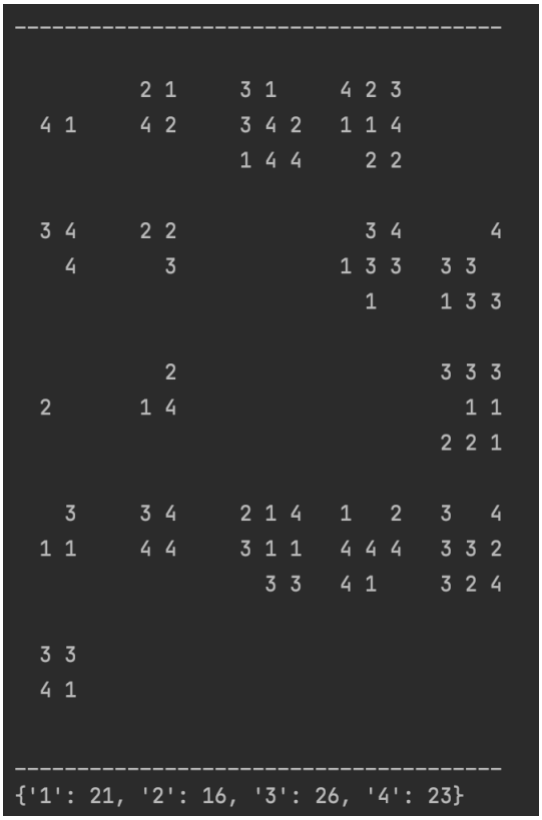
```
 ------------------------------------


           2     4 4 1
   1       3     4 2
         4   1   3 1 4


     2       4 4   3 1 2
   1 1     2 3 2   3 4 1
         3 1 3       1


     1 1     4 1
   2 4 1   4   4
   4 1 1       3

   3 2     2 2 3
   3 4 3   3   2
     3     3 2 1

   1 1 3   2   4           2 1 2
   3 3 2   2   2           3
   1   3   4 1 4           1   4
 ------------------------------------
 {'1': 23, '2': 18, '3': 20, '4': 18}
```

```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 EL_SHAPE
3 4 EL_SHAPE
4 4 EL_SHAPE
5 4 EL_SHAPE
6 4 EL_SHAPE
7 4 EL_SHAPE
8 4 EL_SHAPE
9 4 EL_SHAPE
10 4 EL_SHAPE
11 4 EL_SHAPE
12 4 FULL_BLOCK
13 4 FULL_BLOCK
14 4 FULL_BLOCK
15 4 FULL_BLOCK
16 4 FULL_BLOCK
17 4 FULL_BLOCK
18 4 FULL_BLOCK
19 4 EL_SHAPE
20 4 FULL_BLOCK
21 4 FULL_BLOCK
22 4 FULL_BLOCK
23 4 FULL_BLOCK
24 4 FULL_BLOCK

Done in --- 0.04 seconds ---
```

tilesproblem_1326658928646700.txt



```
------------------------------------
3 4 4 3 1 2 2 4 4 2 2 3 2 3 2 1 1 1   1
4     3   2 1 4 4 3 1   2 4 2 3 1 4 2 1
4 4 1 3 2 4 2 4 4 3 4 2   1 1 4 3 3 2 1
1 3 4   2 3 4 2 4 1 4 4 2   2 2 1     2
4     3 2 4 3 4 1 3 3 1 2 1 1   3 3
  3 4 4 1 2 2 2 1 2 1 4     3 4       4
4   4   2   3 1 4 3 2   2 1 3 3 2 3 3
4 3 3   1 2 3 3 2   3 1 3   1   3 1 3 3
3 1 3 3   2 4 4 3 1 3 2 3 2   1 2     2
3     4 1   2 1 4 4   1 4 1     2 3 3 3
1 2   1   1 4 1   1 4 1 4 2 4 1 3   1 1
3 2   1 3 1 4 2 4 4   2 4   1   1 2 2 1
3 3 3 4 1 4 3 4 2 1       2 1 2 2   2 3
3   3 1 2 3 4 1 4 2 1 4 1 1   2 3 3   4
4 1 1   1 4 4   3 3 1 1 3 4 4 4 1 3 3 2
1 2 3 1         2   3 3 4 4 1   4 3 2 4
3 3 4 1 4 2   4 4 4 3 1 2 2 2 1 2 3   4
  3 3 2 2 2 4   3 2 3   2 3 2 4 4   2 2
4 4 1 1 4   2 4 2 4 3 3 2 1 3 1 1 3 2 2
  2 2 1 3 1 3 1 2 1 4 2   1   3 1 4 2 3
------------------------------------
{'1': 21, '2': 16, '3': 26, '4': 23}
```



```
------------------------------------


        2 1     3 1     4 2 3
  4 1   4 2     3 4 2   1 1 4
                1 4 4     2 2


  3 4   2 2               3 4       4
    4     3               1 3 3   3 3
                            1     1 3 3


          2                       3 3 3
  2       1 4                       1 1
                                  2 2 1


    3     3 4     2 1 4   1   2   3   4
  1 1     4 4     3 1 1   4 4 4   3 3 2
                    3 3   4 1     3 2 4


  3 3
  4 1


------------------------------------
{'1': 21, '2': 16, '3': 26, '4': 23}
```

```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 OUTER_BOUNDARY
3 4 OUTER_BOUNDARY
4 4 OUTER_BOUNDARY
5 4 OUTER_BOUNDARY
6 4 OUTER_BOUNDARY
7 4 OUTER_BOUNDARY
8 4 OUTER_BOUNDARY
9 4 FULL_BLOCK
10 4 EL_SHAPE
11 4 FULL_BLOCK
12 4 FULL_BLOCK
13 4 EL_SHAPE
14 4 FULL_BLOCK
15 4 EL_SHAPE
16 4 EL_SHAPE
17 4 FULL_BLOCK
18 4 EL_SHAPE
19 4 FULL_BLOCK
20 4 FULL_BLOCK
21 4 EL_SHAPE
22 4 EL_SHAPE
23 4 EL_SHAPE
24 4 FULL_BLOCK

Done in --- 7.77 seconds ---
```

tilesproblem_1326658934155700.txt

```
--------------------------------------
  1 4 2 4 2 4 2 1 3 2   1   3   2 2 2 1
4   4 4   1 2   3 3 4   4 2 1 4 2 3 4 2
2 4 2 2 1 2 3 4   2 3 2 1 3 3 4 1 2 1 3
    4 1   4 2 1 3 3 2   3 1           4
4 1 2 2 2 3 3 4 3 1 3 4 2 1 4 3 3   2 1
1 1 2   1 2   4   3 2 1 1   2 2 2 4   4
4 4   4   4 3 2     2 1 4   4 3 4 4 3 4
1   3 2 1 4 4 2 4 1 2 3     4   4   2 1
    4 1 4 3 4 4   3   1 2 2 2 1       2
3   1 2 1 4 2 2 3 4 2 2 3 1   4   1   3
2 3   1 2   2 3 4   4 1   2   1 4 3 3 3
3 1 3 2 3 1 2 2 1   1 1 2 1 1 1 2 1   2
1   4 2 2 3   3 3   1 3 2 1 2   3 3 3 2
1 1 3 4 3 1 2 2 2 1 3 4 3 4 1 1 3 4 1 3
4   1 1 3 4 1 2   4 4 1 1 1 1 1 2     3
1 1   1 4 2 4 3   1 3 3 4 2 1 2   2 1 1
  1 1 1 1 1 2 1 4   4   4 1 3 4 2 4 1 4
2 3 1 4 3 2 3 1 1       2 1 1 3 1 3 2 1
3 4 3 3 1 3 2 4 2 2     3   3 4 4 1 2 4
  4 1   4 2 2 1 3   4 2 4 4 2   1     3

--------------------------------------
{'1': 24, '2': 24, '3': 16, '4': 15}
```

```
--------------------------------------

   4     1 2     3 4             3 4
 4 2     2 3     2 3             2 1


 1 2     2       3 2
 4       4 3       2


   1     4 2     4 2 2   1
 3         2       4 1   2
                   1 1


 1 3     1 2             4 1 1   4 1 3
   1     4 1             1 1 1       3
                         2 1 2   2 1 1


 3 1     2 3                     3 2 1
 4 3     3 2     2               1 2 4
                   4 2                 3
--------------------------------------
{'1': 24, '2': 24, '3': 16, '4': 15}
```

```
# Tiles:
0 4 OUTER_BOUNDARY
1 4 OUTER_BOUNDARY
2 4 OUTER_BOUNDARY
3 4 OUTER_BOUNDARY
4 4 OUTER_BOUNDARY
5 4 OUTER_BOUNDARY
6 4 OUTER_BOUNDARY
7 4 OUTER_BOUNDARY
8 4 OUTER_BOUNDARY
9 4 OUTER_BOUNDARY
10 4 OUTER_BOUNDARY
11 4 OUTER_BOUNDARY
12 4 EL_SHAPE
13 4 FULL_BLOCK
14 4 EL_SHAPE
15 4 FULL_BLOCK
16 4 FULL_BLOCK
17 4 OUTER_BOUNDARY
18 4 EL_SHAPE
19 4 FULL_BLOCK
20 4 OUTER_BOUNDARY
21 4 FULL_BLOCK
22 4 FULL_BLOCK
23 4 EL_SHAPE
24 4 EL_SHAPE

Done in --- 12.74 seconds ---
```