What did we do?

Tenniel Miao, Spencer Robinson, Richard Wohlbold

May 21 2018

Problem description (1)

- **Input**: Terrain matrix
- **Intermediate step**: Interpolated terrain matrix
- **Problem**: Navigating the terrain with a robot
- Different tiles:
  - Normal terrain
  - Fog (hides the tile under it)
  - Pit (impassable; not interpolated)
  - Mountains (impassable; get interpolated)

Problem description (2)

1	1	0	1	1	1	1	1	0
0	1	4	1	2	4	1	1	1
1	1	1	2	1	1	4	1	1
1	2	2	4	1	1	1	1	2
1	1	1	1	3	1	1	1	2
0	1	1	1	1	1	3	1	1
0	1	3	1	1	2	1	1	1
1	0	2	0	1	2	1	1	1

Problem description (3)

1	1	0	1	1	1	1	1	0
0	1	4	4	4	4	4	1	1
1	1	4	4	4	4	4	1	1
1	2	2	4	4	4	1	1	2
1	1	1	3	3	3	1	1	2
0	1	3	3	3	3	3	1	1
0	1	3	3	3	2	1	1	1
1	0	2	0	1	2	1	1	1

Creating the terrain

- Different ideas, pretty complicated in practice
  1. Finding the convex hull of the points
  2. Drawing the lines
  3. Filling all the tiles in the polygon
- Right now: Using scipy.ConvexHull, currently working on our own working solution

Searching for the goal

- Different search algorithms:
  1. Depth-First Search
  2. Follow-Side (Left or Right)
  3. Different random algorithms
- Breadth-First Search not applicable → too much backtracking → inefficient
- Code is very modular → easy to add new search algorithms

Output

- Different output options:
  1. Plain or colored output
  2. With or without ncurses
  3. Rudimentary GUI output
  4. No output
- Code is very modular → easy to add new output methods

Demo

- (Live demo)
- Code under https://github.com/poshut/rst

Conclusion

- Very interesting to work on a project as a team
- Algorithmic challenges
- Software engineering challenges
  - Code readability
  - Code maintainability
  - Consensus on how to write code
  - Writing tests

Thank you for your attention!