IKI10100I: Data Structures & Algorithms 2015-16
Faculty of Computer Science, Universitas Indonesia
Tutorial 3 (Week 4): Recursion, Find F.R.A.N.C.I.S
Deadline: Monday, 29th February 2016 18:00 WIB

1. Problem Description

Wade Wilson is an antihero who has just recently lose his pretty face because of his involvement with Weapon X and someone named Francis. He wants to track him down and 'teach him a lesson'. But Francis is not alone, he is accompanied by Angel Dust, who possesses superhuman levels of strength, stamina and speed. He needs to be careful and avoid her presence. Fortunately, his best friend Weasel obtained a dungeon map that can help him to capture Francis alive!

He gave Wade a dungeon map encoded in ASCII symbols, asked him to find out two things: firstly, Wade must find out whether the entire dungeon can indeed be explored from the starting position or not (for instance, there might be some areas that are blocked off by Angel Dust). Second, Wade must find out how many pieces of clues he is able to recover, and how many pieces of clues there really are.

The problem is: Wade cannot read ASCII symbols, and might need your help to translate that to retrieve clues regarding Francis' position. When you looked at the map and listened to Wade's story, you immediately realized that her problem is one that can be solved using recursion.

The idea is simple: assuming that Wade is currently in a particular cell, he wants to ask the following questions: *"Can I reach every other cell from here?"* and *"How many clues can I retrieve if I start from this point?"*. You realize that the answer to this question can be determined by first asking the same questions in adjacent cells (i.e. cells above, below, to the left, and to the right of the cell he is currently in). A source code template is provided to help you focus on developing the algorithm for exploring the dungeon.

2. Input

The input will be given from a text file. The input file consists of several lines of string with 4 possible symbols that describes the layout (map) of the dungeon. The description for each symbol is as follows:

- 'W' is Wade starting position in the dungeon.
- '.' is an empty space in the dungeon that can be walked through by Wade.
- 'X' represents an obstacle (Angel Dust) that cannot be explored by Wade.
- 'C' represents a clue in the dungeon. Wade can walk through the treasure.

3. Output

The output will be written to the standard output and consist of two string lines.

The first line is a string message that tells whether she can visit all possible empty spaces in the map or not. The format is simply "Wade can explore all areas in the

dungeon. :-)" if he is able to explore the whole dungeon. Otherwise, the format is "Wade cannot explore all areas in the dungeon. :-(".
The second line is a string with the format "Clues obtained: X / Y" where X is the number of clues that can be reached from Wade's starting position and Y is the actual total number of clues in the dungeon.

4. Sample Input and Output

Below are samples of input and output for the maps. In this tutorial, there are three test cases given (1.in, 2.in, and 3.in).

Input: file 1.in

```
XXXXXXX
X.CX..X
XXXX..X
CC....W
XXXCXXX
```

Output:

```
Wade cannot explore all areas in the dungeon. :-(
Clues obtained: 2 / 4
```

Input: file 2.in

```
XXXXX
XXCCX
XWX.X
XXXXX
```

Output:

```
Wade cannot explore all areas in the dungeon. :-(
Clues obtained: 0 / 2
```

Input: file 3.in

```
...
W..
CCC
```

Output:

```
Wade can explore all areas in the dungeon. :-)
Clues obtained: 3 / 3
```

5. Submission

Compress your .py files into a ZIP file named according to the following file naming format: **`YourNPM_YourName_Tutorial2.zip`**. Upload the ZIP file to the provided submission slot in SCeLE. The submission deadline is February 29th 2016 18:00.

Good luck!