CSCI 1100 — Computer Science 1
Homework 9
ASCII Font Art

Overview

Homework 9, worth **100 points** towards your homework grade, is due by **11:59:59 pm on Thursday, December 4, 2014.** All submission rules, academic integrity rules, and late penalties apply. Please pay careful attention to the submission instructions at the end of this document.

In this homework, you will write a program that takes as input a sentence from the user, characters to use for the background and foreground of the ASCII art output. You will then print the given sentence using ASCII art.

To accomplish this task, you will read a file called `simple_font.txt` which contains an ASCII art version of each character. The file format is described below.

To solve this homework, you can use any method we have learnt in this class. Feel free to discuss with us or others in the class what the appropriate data structures are to solve this.

Font File Format & Parsing of the Font File

The `simple_font.txt` file contains a fixed width "bitmap" representation of each printable character. In this file every letter uses '`#`' for the foreground character and '`.`' for the background.

The file has one line of header. Then, each character is given in eight lines. The first line contains the character ASCII code, followed by the character. Then, the next 7 lines is the ASCII art version of the character.

Here are the lines corresponding to the header and the ASCII art for `S`:

```
83 'S'
.####.
#.....
.####.
.....#
#....#
.####.
......
```

Kerning

Your program must do kerning. What is it you ask? It is a way to make sure that each character appears the same distance from each other.

Each character in the input file is exactly 6 characters wide. However, some characters are too wide, with lots of fillers on either side. The idea of kerning is to change each character so that there is only one space (`.`) vertically before after the actual character.

If you used the kerned version of all the characters, the produced text will look more pleasing to the eye.

As an example, take a look at `i` for example before and after kerning:

```
.#....                        ..#..
......                        .....
##....                        .##..
.#....                        ..#..
.#....                        ..#..
###...                        .###.
......                        .....
```

Before kerning   After kerning

You can learn more about it here:

`http://en.wikipedia.org/wiki/Kerning`
`http://type.method.ac/`

Note that space character is not changed due to kerning.

Deliverables

Turn in a single file called `hw9.py` that reads three values:

- input sentence
- foreground character (newline to keep `#`)
- background character (newline to keep `.`)

print each value after reading. Your program must assume the existence of the file `simple_font.txt`, and output:

- ASCII art using your input foreground/background characters without kerning
- The same ASCII art now with kerning (this last part is only 15 points, but highly recommended for those of you considering taking more advanced Computer Science courses).

Sample output is provided in the ZIP folder for this homework.

A word of advice: It is best to complete parsing the input file first. Then, check your parsing with single letter inputs. Next, add multi character inputs to your program. And, finally add kerning. Also, look at the output of your program from afar for best results!