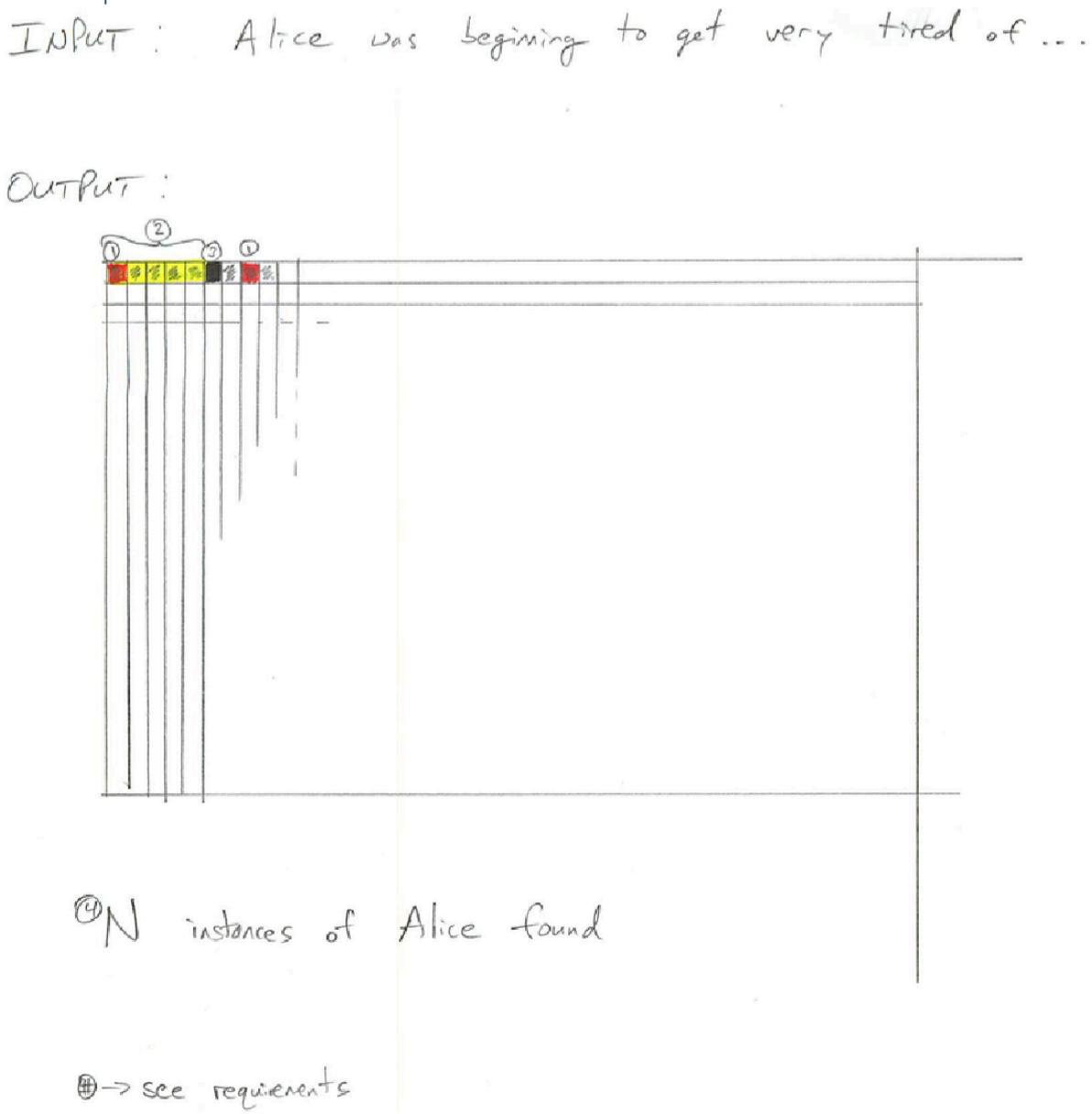
Project 1

Parsing and Visualizing Unstructured Data

Create a visualization of Lewis Carroll's Alice in Wonderland, using Java and the Processing environment.

Screen Specification



Requirements

1. Each instance of a particular character should be colored the same. The colors are up to you, and this should not be case sensitive. For example, all **A**'s and **a**'s should be chartreuse.
2. Each occurrence of the word **Alice** should be highlighted in some fashion.
3. Spaces and non-alpha characters should all be the same color (eg. black).
4. The program should display the number of times that **Alice** occurs in the text.
5. When the user clicks on the screen, the program should display a simple visualization that communicates the frequency of each character in the text. This second visualization should also communicate the least frequent character, and the most frequent character.
6. Your program should treat the data as a stream, and use a *java.io.BufferedReader*.

Learning Outcomes

- Imagine data as a stream of bytes.
- Recite the relationship between files, buffers and streams.
- Apply the Java language and the Processing graphics API to visualize a piece of unstructured data.
- Relate characters to their numeric representation.
- Create a working program given a specification and set of requirements.
- Investigate frequency, minimum, maximum and how they manifest as code.
- Decide how to apply aesthetics and visual design to communicate information.
- Investigate technical documentation to apply unfamiliar APIs to accomplish a goal.
- Apply data structures to manage the state of a running program.

Assessment

Uses a buffer to contain the input data	10
Treats upper and lower case letters the same	10
Ignores non-alpha characters	10
Consistently colorizes character occurrences	10
Draws visual attention to occurrences of keyword (Alice)	5
Displays correct frequency of occurrences of keyword	5
Initial visualization of the input data	10
Program reacts to user input, and displays second visualization	5
Second visualization organizes and displays the letter frequencies	5
Second visualization draws attention to the least & most frequent characters	5
Program correctly calculates frequencies	5
Program uses appropriate data structures for managing data	5
Program is well written, adhering to best practices in writing and design	10
Program contains comment header with name, project # and description	5
Program posted to GitHub and demonstrates a commit history of work	+5
Total	100

Extension & Additional Challenges

Investigate the format of an mp3 file including ID3 headers. Parse the components of the file, and visualize those components in a manner that reflects the internal encoding format of the mp3 specification. Parse the text in the ID3 header and display this text on the screen as well.