# Program Documentation

The program works by parsing a text document, searching for the key-words: “text”, “image”, “prompt”, “sim”, “pause”, “ln”, “f”, and “othertext”. Each line will have a keyword, and may also have parameters after this. main.py is responsible for interpreting and executing the text document, called mainData.txt.

The reason I developed an interpreter and data rather than just hard-coding the entire program is because this simplified the challenge. I am now not entirely convinced this did make the program easier, but I didn’t have time to go back on it, and I do think it’s kind of neat.

The mainData document runs the program by calling formulaically named text documents, as well as very modular “simulations” which are essentially just separate python problems specifically designed to test some cryptographic concept. Every time this happens, there is a password that needs to be found, which is hashed and compared to pre-hashed values in the hash\_values folder, before the program continues.

Originally, I was going to encrypt all but the first text document, and base their decryption keys on the passwords gained from the cryptography tasks, in order to stop players from skipping ahead. Sadly, I could only implement encryption using random keys, and was not sure how to implement strong encryption based arbitrary strings. It would also just be a bit over the top.

I didn’t have time to debug the entire program, so there is lots of empty error catching just to hope that if the program crashes once I have submitted it, it’ll just skip some content and keep on going without too many people noticing.