Brute Force Password Cracker CS111

Isaac Earnhart

Project Proposal

- A few miscellaneous proposals
- Password cracker for mass table of passwords and usernames
- Options for which characters to use
- Different hash methods

Implementation

- Brute Force Password Cracking
- Options for different characters sets
- Storing password and username

Future Developments

- Easily implement new hashing method with new class
- Options for starting with Capital letter, ending with number
- Stream files in order to crack whole databases

DEMO

Console Input w/ Error Checking

- Cs111 class
- Valid minimum and maximum password length

Console Output

- Output prompts for user entered data

Formatted Output

- Individual class toString method for outputting onto screen

Selection Statements

- Options for including symbols, numbers, capital letters

Advanced Boolean Expressions

- BruteForce method in HashAlgorithms

Repetition Statements

- bruteForce method and incrementChar method in HashAlgorithms

Classes/Objects

- HashAlgorithms
- Individual

Static Variables and Methods

- Interface
- Cs111

Math Class

- Individual Default Constructor chooses a random name in none are given

Wrapper Classes

- Cs111.readInt uses Integer.parseInt

References (Deep Copying)

- Copy constructor in HashAlgorithms and Individual

Arrays / Multi-Dimensional Arrays

- incrementChar and bruteForce methods

Inheritance

- SimpleHash derivative of HashAlgorithms

Polymorphism

- HashAlgorithms method hash(String password) overridden in SimpleHash
- Used in HashAlgorithms within bruteForce

File Input/Output

- Outputting the start screen from text file