

Monday

Monday's topics:

- Subject of Cryptography
 - Defining Cryptography
 - What it means to be "Cryptographically secure"
 - What makes a good cipher
 - Different ways to break a cipher
- Levels of programming (Machine, Assembly, Compiled, Interpreted)
- Intro to C++
 - Input and Output
 - Conditionals
 - While and For loops
- Shift Ciphers

Morning

Lecture 1: C++

- Location on program stack (compiled language)
- Input & Output
- Conditionals

Programming Time

- Can use game plan to practice programming through conditionals
- If they finish ahead of time, move them towards loops
- Can write a basic mad libs or choose your own adventure as practice

Afternoon

Lecture 2: Cryptography

- Define cryptography
- Discuss how to make a good cipher (Have students come up with potential ideas)
- Define Cryptographic Security

Activity: Shift Ciphers

- Discuss shift ciphers in an interactive activity
- Demonstrate an actual, implemented shift cipher

Programming Time

- Continue to work up to for loops
- Advanced students can begin to make their own shift cipher:
 - Write the requirements on a whiteboard or similar
 - Break down the program into easily accomplished steps (input, encryption, output, etc)

Tuesday

Tuesday's Topics:

- C++, cont. (If many students still need time)
- Hashing
 - Defining hashing
 - Why we use hashing (Use password storage as an example)
- Intro to Python
 - Input and Output
 - Conditionals
 - While and For loops

Morning

Lecture 3: Python

- Location on the program stack (Interpreted Language)
- Input and Output
- Conditionals
- Discuss differences with C++

Programming Time

- Use the python game plan to start learning python (again, working towards loops)
- Students should be able to write a choose your own adventure or mad libs program once again

Afternoon

Lecture 4: Hashing

- What hashing is
- Discuss - Why do we use hashing?
- Talk about password storage
- Attempt to design a basic hashing algorithm
- Show them your implementation of MD5

Programming Time

- Set up pyCrypto library (You should download it ahead of time)
- Continue work with python
- Advanced students should experiment with the hashing and encryption available in pyCrypto
 - Try to make a password system

Wednesday

Wednesday's Topics:

- Binary
 - What it is
 - How it's used to represent different data types
 - What it means in the context of encryption
 - Most encryption is bitwise
- Symmetric and Asymmetric Encryption
 - Definition
 - XOR Encryption
 - How SSL Works
- Start Projects!

Morning

Lecture 5: Binary

- What it is
- How to represent different data types with binary
- Operations on binary (AND, OR, NOT, XOR)
- Application to encryption (NOT gates)

Programming Time

- Work on either shift ciphers, XOR ciphers, hashing, or implementing pyCrypto encryption
- Begin to discuss final projects
- Optionally students can continue studying binary with the commodore 64 game plan

Afternoon

Lecture 6: Symmetric and Asymmetric Encryption

- Discuss the two kinds of encryption: Symm. and asymm.
 - Symm. uses one key
 - Asymm. Uses a public and private key
- Talk about how SSL encryption works (using both symm. and asymm. encryption)
- Discuss XOR ciphers and how they work
- Show implementations (depending on interest) of:
 - XOR ciphers
 - Blowfish
 - RSA

Programming Time:

- Decide on final projects
- Students can use either python or c++
- Implementations of pycrypto, actual encryption types, or a custom kind of encryption encouraged