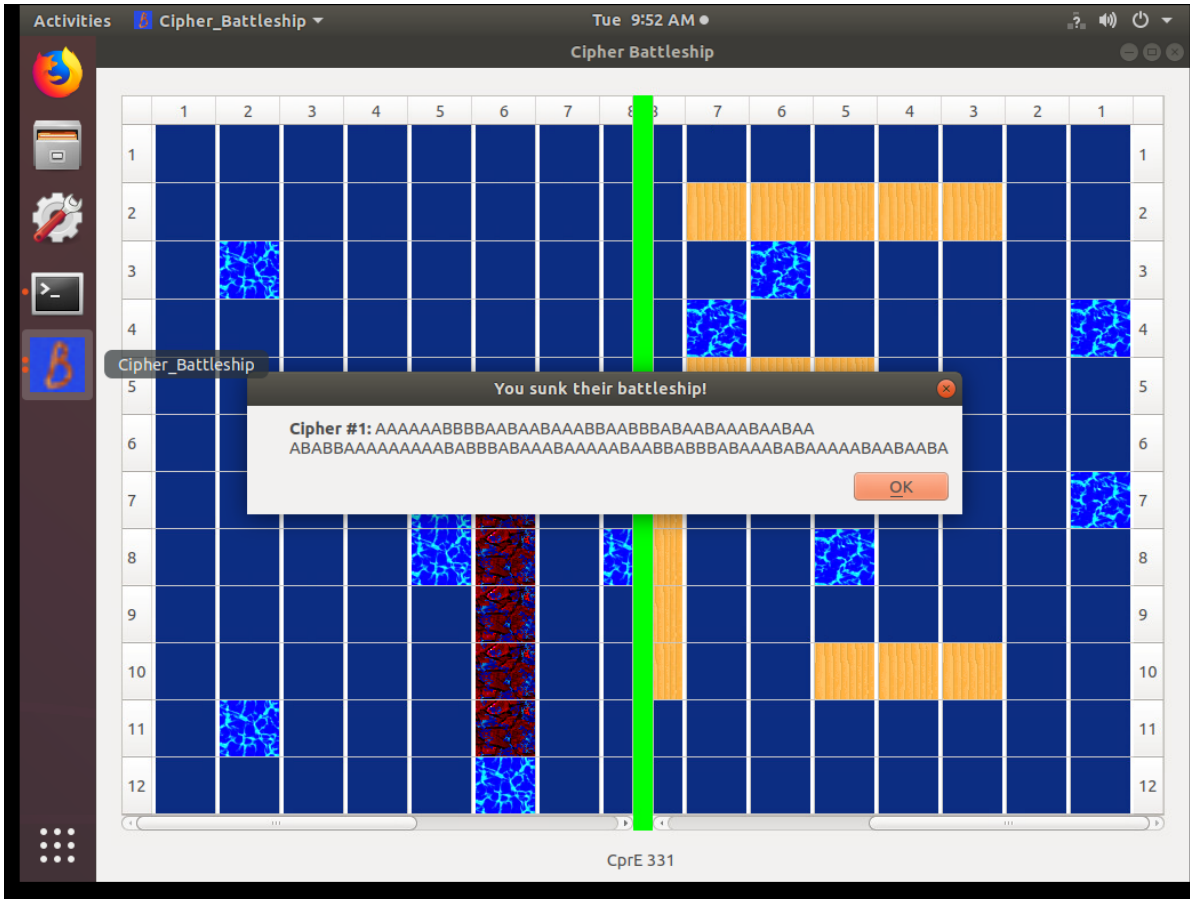
1) **Cipher 1:**
   a. **Screenshot of the board with ciphertext displayed**
   (4 points)

   Cipher #1: AAAAAABBBBAABAABAAABBAABBBABAABAAABAABAA
   ABABBAAAAAAAAABABBBABAAABAAAAABAABBABBBABAAABABAAAAABA
   ABAABA



   b. **Name of the classic cipher**
   (2 points)
   Bacon's Cipher

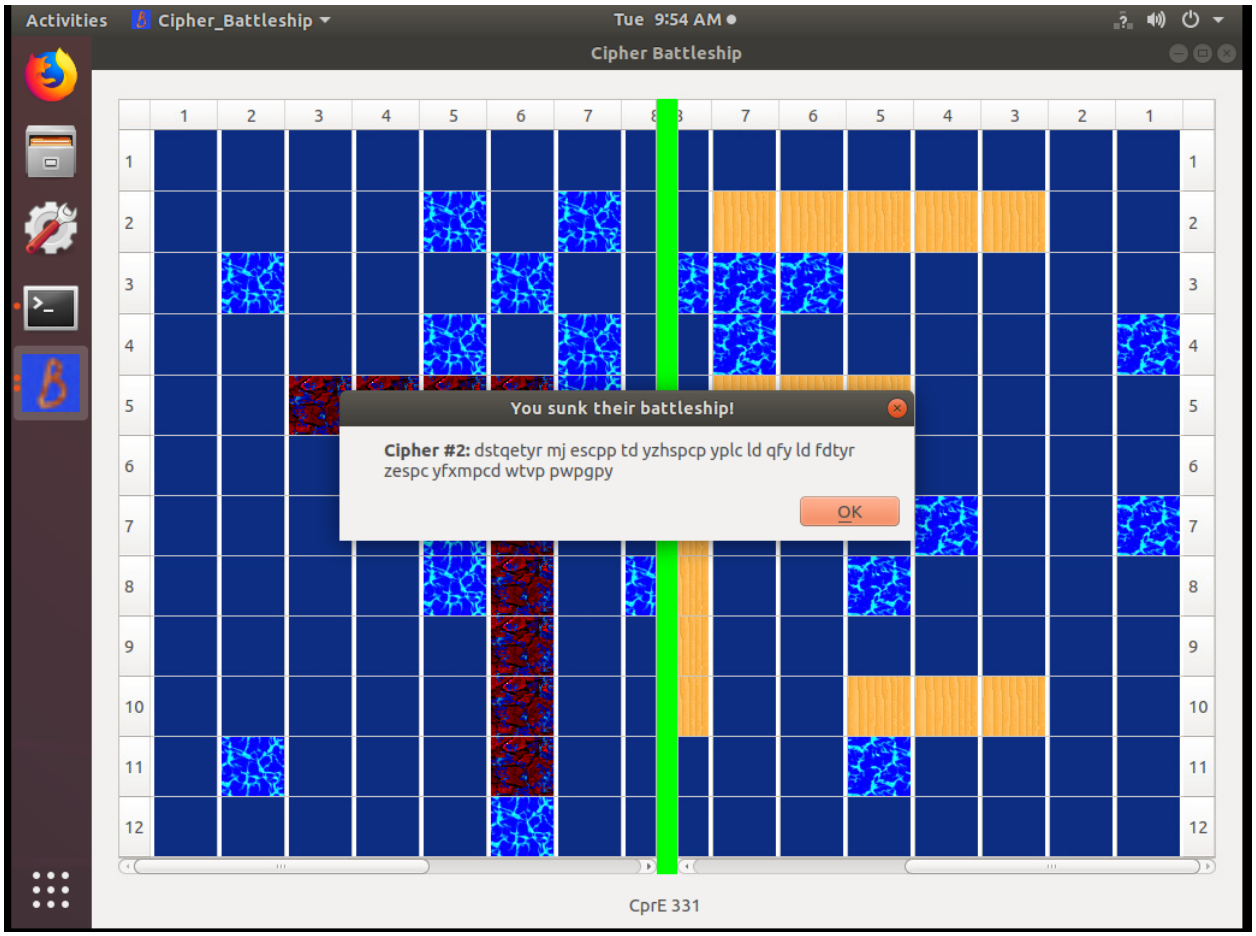
   c. **Key**
   (2 points)
   AAAAA = "a", AAAAB = "b", AAABA = "c" …. BBAAB = "z"

   d. **Plaintext message**
   (4 points)
   APERTURELABORATORIES

2) **Cipher 2:**

a. **Screenshot of the board with ciphertext displayed**
(4 points)

Cipher #2: dstqetyr mj escpp td yzhspcp yplc ld qfy ld fdtyr zespc yfxmpcd wtvp pwpgpy



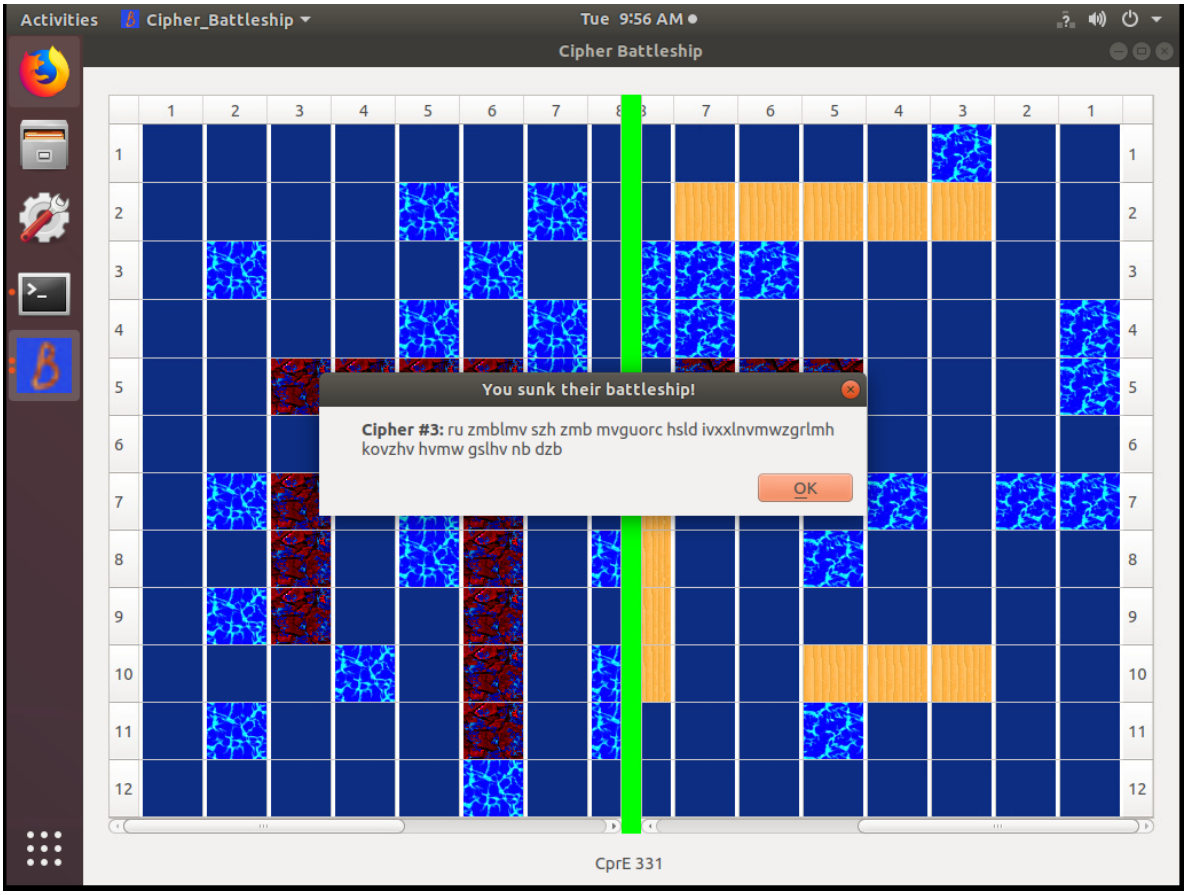
b. **Name of the classic cipher**
(2 points)
Caesar Cipher

c. **Key**
(2 points)
Shift forward by 11 (A == L and S == D)

d. **Plaintext message**
(4 points)
shifting by three is nowhere near as fun as using other numbers like eleven

3) **Cipher 3:**

a. **Screenshot of the board with ciphertext displayed**
(4 points)
Cipher #3: ru zmblmv szh zmb mvguorc hsld ivxxlnvmwzgrlmh kovzhv hvmw gslhv nb dzb



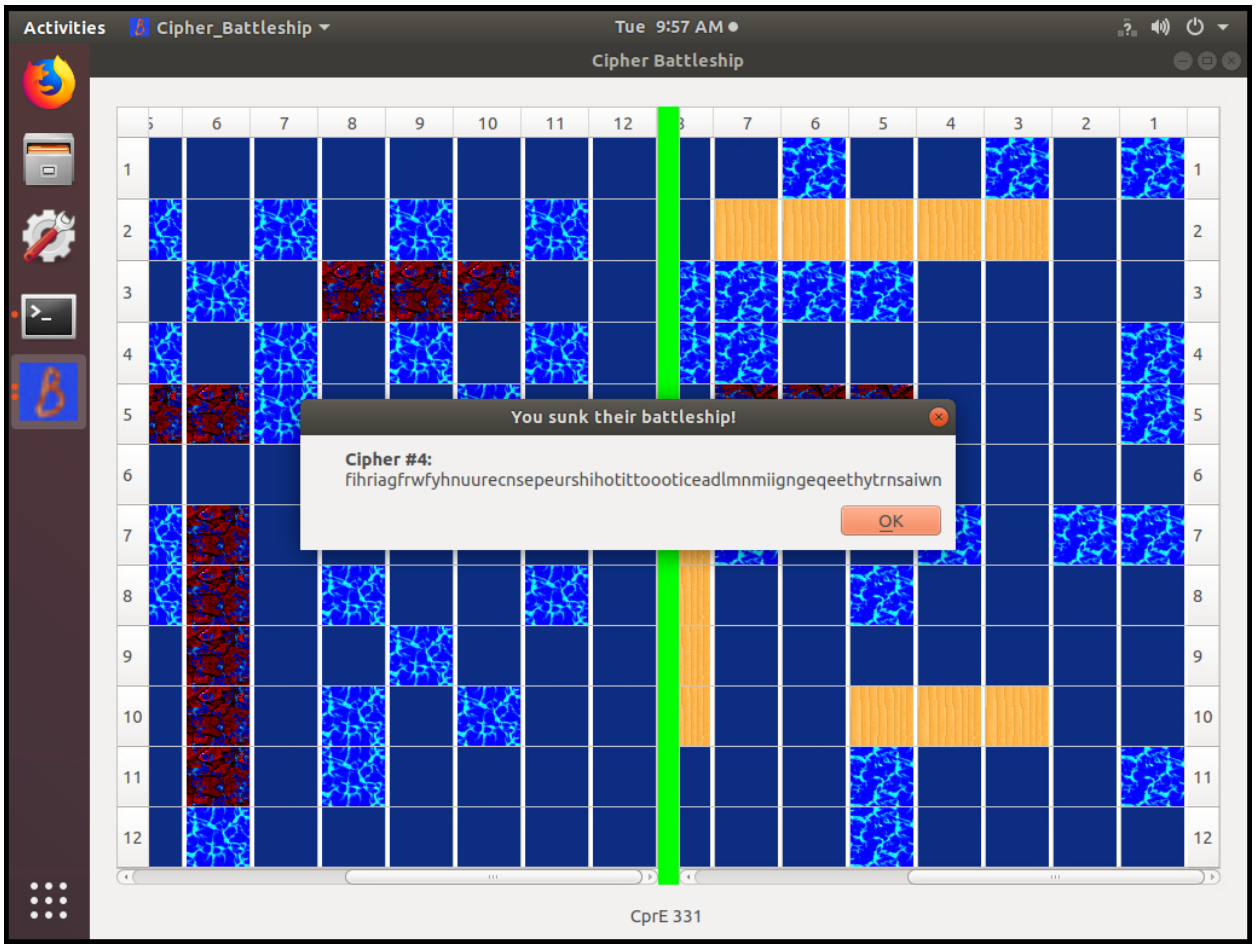
b. **Name of the classic cipher**
(2 points)
Atbash

c. **Key**
(2 points)
Flip the alphabet in reverse (A = Z, M = N, and S = H)

d. **Plaintext message**
(4 points)
if anyone has any netflix show reccomendations please send those my way

4) **Cipher 4:**
   a. **Screenshot of the board with ciphertext displayed**
      (4 points)

Cipher #4: fihriagfrwfyhnuurecnsepeurshihotittoooticeadlmnmiignqeqeethytrnsaiwn



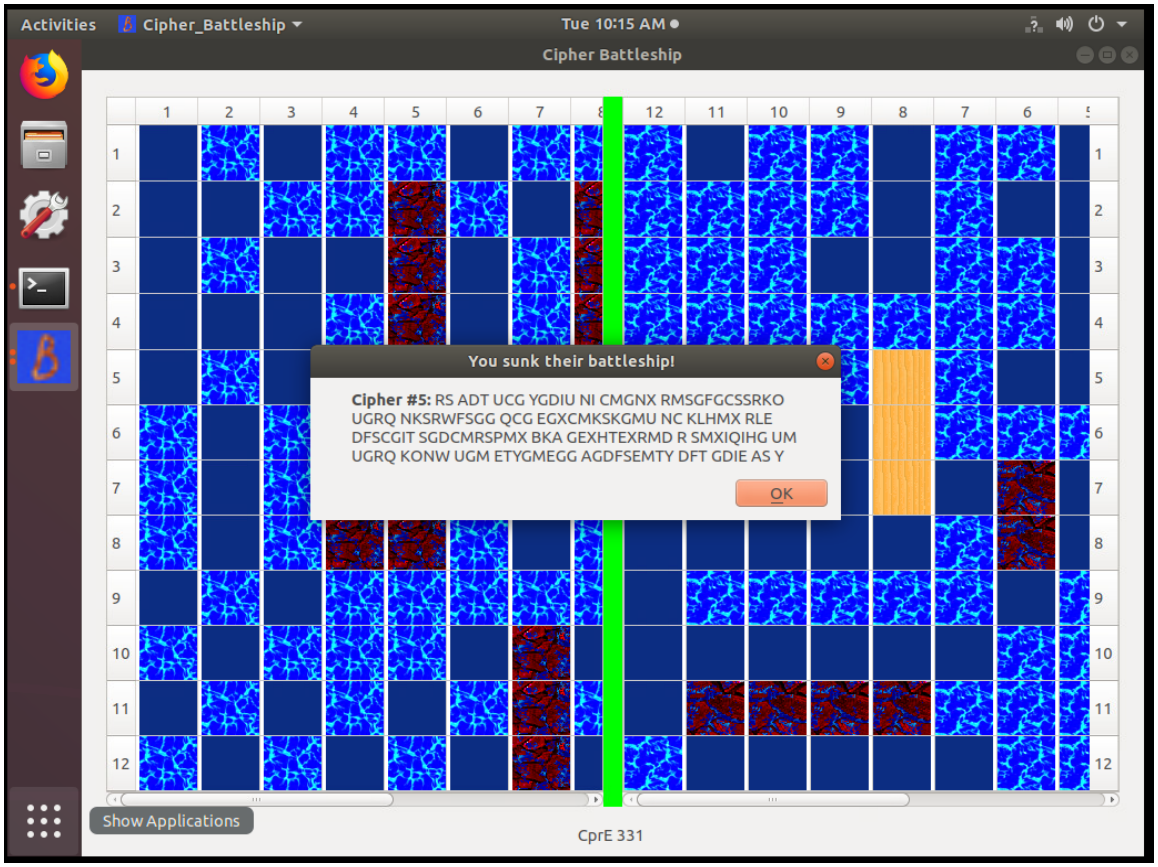
   b. **Name of the classic cipher**
      (2 points)
      Rail Fence (Zig-zag)

   c. **Key**
      (2 points)
      Message is read diagonally of the text going top to bottom then bottom to top, repeating

   d. **Plaintext message**
      (4 points) (No spaces)
      FENCING SHEEP REQUIRES A HEIGHT OF THIRTY TWO TO FORTY INCHES AND ALUMINUM WIRING

5) **Cipher 5:**

a. **Screenshot of the board with ciphertext displayed**
(4 points)
Cipher #5: RS ADT UCG YGDIU NI CMGNX RMSGFGCSSRKO UGRQ NKSRWFSGG QCG EGXCMKSKGMU NC KLHMX RLE DFSCGIT SGDCMRSPMX BKA GEXHTEXRMD R SMXIQIHG UM UGRQ KONW UGM ETYGMEGG AGDFSEMTY DFT GDIE AS Y



b. **Name of the classic cipher**
(2 points)
Play Fair

c. **Key**
(2 points)
“Fair” w/o J

d. **Plaintext message**
(4 points)
IT WAS THE TREAT OF ENEMY INTERCEPTION THAT MOTIVATED THE DEVELOPMENT OF CODES AND CIPHERS TECHNIQUES FOR DISGUISING A MESXSAGE SO THAT ONLY THE INTENDED RECIPIENT CAN READ IT X

6) **New "classic" cipher**

BoiCipher:

Shift cipher but with a twist.
Each character in the text is shifted by
(index of text + 1) * (shift value)
and the text is flipped beforehand

Usage:

-h: print help information
-e: encrypt text (needs -t and -k set)
-d: decrypt text (needs -t and -k set)
-k: the shift value (integer)
-t: text to encrypt/decrypt (string)

a. **Function to encrypt any given plaintext**
(15 points)
Made encrypt(text : str, key : int) -> str in BoiCipher.py
Can be used via command line with -e option

Ex:

```
❯ ./BoiCipher.py -e -k 115 -t "How is it going"
Encrypted text: rjpgjhhcdcfj
```

b. **Function to decrypt a given ciphertext**
(15 points)
Made decrypt(text : str, key : int) -> str in BoiCipher.py
Can be used via command line with -d option

Ex:

```
❯ ./BoiCipher.py -d -k 115 -t "rjpgjhhcdcfj"
Decrypted text: howisitgoing
```

c. **Follows specifications**
(5 points)

d. **Function name**
(5 points)