

Object Oriented Caesar Cipher

Practice Quiz, 4 questions

1
point

1.

Which of the following options is the best choice for adding an additional private field to the **CaesarCipher** class to make it easier to call the **decrypt** method on a string that was encrypted using an object of this class?

- ☐ The character variable that reads one character at a time from the input String parameter in the **encrypt** method.
 - ☐ The String **input** which is a parameter to the **encrypt** method.
 - ☐ The integer **key** which is a parameter to the constructor.
 - ☐ The for loop variable in the for loop in the **encrypt** method.
-

1
point

2.

Which one of the following best describes the approach for the **decrypt** method that has one String parameter **encrypted**?

Assume **decrypt** can also access the **key** parameter that was used in the constructor, and that value is stored in an instance variable named **mainKey** in the constructor.

- ☐ Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(mainKey);  
2 return cc.encrypt(encrypted);
```

- ☐ Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(26);  
2 return encrypt(cc.encrypted);
```

- ☐ Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(26);  
2 return cc.encrypt(encrypted);
```

- ☐ Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(mainKey);  
2 return encrypt(cc.encrypted);
```

Object Oriented Caesar Cipher

Practice Quiz, 4 questions



Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(26-mainKey);  
2 return cc.encrypt(encrypted);
```



Two lines are needed:

```
1 CaesarCipher cc = new CaesarCipher(26-mainKey);  
2 return encrypt(cc.encrypted);
```

1
point

3.

Which one of the following is the best idea for the method **breakCaesarCipher**?

- ☐ Calculate the frequency of all the letters using **countLetters** and compute the index of the largest frequency using **maxIndex**. Use those values to determine the key, then create a **CaesarCipher** with that key and call **decrypt** on the encrypted string.
- ☐ Create a **CaesarCipher**. Then calculate the frequency of all the letters in the encrypted string, using **countLetters** and compute the index of the largest frequency using **maxIndex**. Then call **decrypt** on the encrypted string.
- ☐ Compute a **CaesarCipher** object for every possible key. Then for each one, decrypt and then calculate the frequency of all the letters using **countLetters**. Compute the index of the largest frequency over all of them. Return the decrypted string that goes with **maxIndex**.
- ☐ Compute a **CaesarCipher** object, and then call **encrypt**.

1
point

4.

In the class **TestCaesarCipher**, should the method **countLetters** be public or private?

- ☐ public
- ☐ private

Submit Quiz