Following what you have learned about the Caesar cipher and using a key=23, how would you encrypt the following quote?

"Programming can be fun, so can cryptography; however they should not be combined" *-Kreitzberg and Shneidermann*

Note: Pay particular attention to the differences in the encrypted words in bold font.

- **MOLDOAJJFKD ZAK YB** CRK, PL **ZAK ZOVMQULDOAMEV;** ELTBSBO QEBV PELRIA **KLQ** YB ZLJYFKBA
- **MOLDOXJJFKD KLQ YB** CRK, PL **ZXK ZOVMQLDOXMEV;** ELTBSBO QEBV PELRIA **ZXK** YB ZLJYFKBA
- **MOLDOXJJFKD ZXK YB** CRK, PL **ZXK ZOVMQLDOXMEV;** ELTBSBO QEBV PELRIA **KLQ** YB ZLJYFKBA

Correct

Continue

Introduction

Have a question? Discuss this lecture in the week forums.

Interactive Transcript

Search Transcript English

0:03
Welcome back. Now that you know a bit about the importance of cryptography, it is time for you to learn a bit more about the concepts of the Caesar Cipher which you will implement in this lesson.

0:14
Suppose you are in a battle and you wanted to send a message to your sub commanders to tell the first legion to attack the east flank.

0:23
You don't want your enemies to know your plan even if they intercept this message. So, you encrypt it with your cipher as shown on the second line.

0:32
The Caesar Cipher algorithm is named for Julius Caesar, the famous Roman emperor who used it.

0:39
The basic idea of this algorithm is to substitute each letter with the letter obtained by shifting the alphabet by a fixed amount. That is a specific number of letters later in the alphabet.

0:52
The amount you shift the alphabet by is the key for this cipher. Julius Caesar used a shift of three letters prior. Which if you think of the algorithm in terms of shifting to a later letter will be the same as 23 letters later. To see how this algorithm works we'll walk through an example of encrypting this message. We'll use the alphabet to show how letters are encrypted. The first letter of the message is F. We find the letter F in the alphabet here and then we go backwards three letters, E, D, C. So you would write down a C as the first letter of your message. The next letter is I. We find the letter I in the alphabet here and then we go backwards three letters, H, G, F, writing down F as the next letter of the message. The next letter is R. We find the letter R in the alphabet here. Again, go backwards three letters, Q, P,O writing down O as the next letter of the message. You would continue the same way through the first word and then, you would get to a space. Doing this by hand as Caesar would have done, the easiest thing to do is leave the space unchanged and write down a space in your message.

2:03
The next word in space proceed in the same way. However, what happens when you get to A? We find A here, it is the first letter in the alphabet. But how do you go three letters backwards? You have to wrap around to the end of the alphabet. From there you go three letters backwards to Z, Y, X, writing down X as the next letter. You continue through the rest of the message in the same way! And end up with something that is unintelligible under casual scrutiny.

2:34
However, if you know, or can figure out the key. You can decrypt the message. The process is the same as encrypting, with a key of 26 minus N.

2:46
So, how do you actually do this? One way: math on letters. If you took our Coursera course, programming, and the web for beginners, you should remember that everything is a number. If you're not familiar with this concept, it is very important in computer science, as computers can only work with numbers. In this case, that principle says that these letter are actually represented as numbers. So you can do math on them.

3:13
In particular, you could tell Java to subtract 3 from the letter F and it would compute the letter C. However, what if you subtracted three from the letter A?

3:24
Java would not know that you wanna wrap around and stay only within the alphabet. So you would have to include some more mathematical operations or a conditional statement to wrap around and get X.

3:37
Another way you could do this, which makes the wrap around case a bit cleaner, is to pre-shift the entire alphabet.

3:44
That is compute the shifts of each letter at the start, before you try to encrypt anything in the message. For example, you could take the alphabet and for a shift of three to the left, computer string like this one. We will see the details of how to do this in a future video. However, once you have computed these strings, You can use them to look up the encryption of each letter. For the f at the start of the message you want to find f in the original alphabet. Think for a moment about what you have learned about strings in the past.

4:17
What method might you use to find F?

4:21
Once you have found F, you look at the letter in the same position in the shifted alphabet which is C. Then you write down that letter in your encrypted message.

4:31
For A which wraps around to X, you do not have any special case. Again, you just find A in the original alphabet, look at the letter in the same position in the shifted alphabet, in this case that letter X. So you write down X in your encrypted message.

4:48
Great, now you know the basic ideas behind a Caesar Cipher. However, before your implement this algorithm, you will need to learn a few new Java concepts. You're going to learn some new ways to manipulate strings. As well as for loops which count over a range of numbers.

5:06
For loops which count over ranges of numbers are particularly important as you will use the numbers you count to index into data, manipulating particular locations in the sequence. You are familiar with strings, which are sequences of characters. But we'll learn about the new types of sequences in the rest of this course. So you will use for loops a lot. Thank you.

Downloads

Lecture Video mp4
Subtitles (English) WebVTT
Transcript (English) txt

Would you like to help us translate the transcript and subtitles into additional languages?