Experiment no. 1: Design & Implement Substitution Cipher

Learning Objective: Student should be able to design and implement Substitution Cipher.

Tools: C/C++/Java/Python or any computational software.

Theory: A substitution cipher is a method of encoding by which units of plaintext are replaced with ciphertext, according to a fixed system; the "units" may be single letters (the most common), pairs of letters, triplets of letters, mixtures of the above, and so forth. The receiver deciphers the text by performing the inverse substitution.

There are number of different types of substitution cipher. If the cipher operates on single letters, it is termed a simple substitution cipher; a cipher that operates on larger groups of letters is termed polyalphabetic. A monoalphabetic cipher uses fixed substitution over the entire message, whereas a polyalphabetic cipher uses a number of substitution at different posistions in the message, where a unit from the plaintext is mapped to one of several possibilities in the ciphertext and vice versa.

The function for Additive/Shift/Generalized Caesar Cipher is given as follows:

- It can use any shift from 1 to 25, i.e., replace each letter by a letter a fixed distance away.
- Ci=E(Pi)=(Pi+k) mod 26 and Pi=D(Ci)=(Ci-k) mod 26.

In Cryptography, a Caeser Cipher also known as Caesar's Cipher, the Shift Cipher, Caesar's Code or Caesar Shift, is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. For example, with the left shift of 3, A would be replaced by D,E would become H, and so on. The method is named after Julius Caeser, who used it in private correspondance.

- The function for Caesar Cipher is defined as follows:
- Ci= E(Pi)=(Pi+3) mod 26.
- Pi=D(Ci)=(Ci-3) mod 26.
- **Example: Plain Text:** ABCDEFGHIJKLMNOPQRSTUVWXYZ
- **Cipher Text:** DEFGHIJKLMNOPQRSTUVWXYZABC

Substitution Cipher can be compared with the transposition ciphers. In a transpostition cipher, the units of the plaintext are rearranged in a different and usually quite complex order, but the unit themselves are left unchanged. By contrast, in a substitution cipher, the units of the plaintext are retained in the same sequence in the ciphertext, but the units themselves the altered.

Source Code:

```
import java.io.*;
import java.util.*;
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class SubstitutionCipher {
    private String alphabet;
    private int key;
    private Map<Character, Character> dict;

    public SubstitutionCipher(String alphabet, int key) {
        this.alphabet = alphabet;
        this.key = key;
        this.dict = generateDictionary();
    }

    private Map<Character, Character> generateDictionary() {
        Map<Character, Character> map = new HashMap<>();
        for (int i = 0; i < alphabet.length(); i++) {
            map.put(alphabet.charAt(i), alphabet.charAt((i + key) % alphabet.length()));
        }
        return map;
    }

    public String encrypt(String plainText) {
        StringBuilder cipherText = new StringBuilder();

        for (char c : plainText.toCharArray()) {
            if (alphabet.indexOf(c) != -1) {
                cipherText.append(dict.get(c));
            } else {
                cipherText.append(c);
            }
```

```
}

    return cipherText.toString();
  }

  public String decrypt(String cipherText) {
    StringBuilder decryptedText = new StringBuilder();

for (char c : cipherText.toCharArray()) {
        if (dict.containsValue(c)) {
          decryptedText.append(dict.entrySet().stream()
                .filter(entry -> entry.getValue().equals(c))
                .findFirst()
                .get()
                .getKey());
        } else {
          decryptedText.append(c);
        }
      }

      return decryptedText.toString();
  }

  public static void main(String[] args) {
      String allLetters =
              "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";
      int key = 5;
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter plain text: ");
      String plainText = scanner.nextLine();

      SubstitutionCipher cipher = new SubstitutionCipher(allLetters, key);

      String cipherText = cipher.encrypt(plainText);

      System.out.println("Cipher Text: " + cipherText);

      String decryptedText = cipher.decrypt(cipherText);

      System.out.println("Decrypted Text: " + decryptedText);

      scanner.close();
  }
}
```

Output:

```java
import java.io.*;
import java.util.*;
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;

public class SubstitutionCipher {
    private String alphabet;
    private int key;
    private Map<Character, Character> dict;

    public SubstitutionCipher(String alphabet, int key) {
        this.alphabet = alphabet;
        this.key = key;
        this.dict = generateDictionary();
    }

    private Map<Character, Character> generateDictionary() {
        Map<Character, Character> map = new HashMap<>();
        for (int i = 0; i < alphabet.length(); i++) {
            map.put(alphabet.charAt(i), alphabet.charAt((i + key) % alphabet.length()));
        }
        return map;
```

```
input
Enter plain text: account privacy has been compromised
Cipher Text: fhhtzsy uwnAfhD mfx gjjs htruwtrnxji
Decrypted Text: account privacy has been compromised


...Program finished with exit code 0
Press ENTER to exit console.
```

Applications:

1. **Education:** Substitution ciphers are often used in educational settings to teach basic cryptography concepts. They provide a simple introduction to encryption and decryption algorithms.
2. **Puzzles and Games:** Substitution ciphers are frequently used in puzzles, games, and treasure hunts. They add an element of challenge and intrigue as participants try to decipher the encoded messages.
3. **Historical Communication:** Throughout history, substitution ciphers have been employed for secret communication. Ancient civilizations, military groups, and spies have used various substitution methods to transmit confidential information securely.
4. **Digital Security:** While basic substitution ciphers are not secure for modern digital communication, they can serve as building blocks for more complex encryption algorithms. Understanding the principles of substitution ciphers can aid in the development of stronger cryptographic protocols.

Result & Discussion:

The experiment entailed creating a substitution cipher using the English alphabet and a randomly shuffled permutation as the key for encryption and decryption. Implementation was conducted in Python, featuring functions for encryption, decryption, and key generation, with thorough testing for accuracy. While the substitution cipher proved simple and efficient for basic cryptographic tasks, its vulnerability to frequency analysis underscored its unsuitability for securing sensitive information without additional security measures or enhancements, suggesting potential future directions for improved cryptographic protocols.

Learning Outcomes: The student should have the ability to design & implement Substitution Cipher

LO1: To describe & understand about Substitution cipher techniques

LO2: To implement Substitution cipher techniques

Course Outcomes: Upon completion of the course students will be able to understand & implement Substitution Cipher.

Conclusion:

In conclusion, the substitution cipher algorithm, employing functions for encryption, decryption, and key generation, offers simplicity and ease of implementation, making it ideal for educational purposes and basic cryptographic tasks. With a time complexity of O(n) for both encryption and decryption and a space complexity of O(26), it proves computationally efficient for small to moderate-sized messages. However, its vulnerability to frequency analysis and fixed mappings pose significant security risks, necessitating additional enhancements like polyalphabetic ciphers for securing sensitive data effectively.

NAME: Aryan Tiwari

CLASS: SE – CSE

ROLL NO: 57

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				