

Step 1: Start

Step 2 : Initialize Variables:

2.1: Define BUFF as 100.

2.2: Declare message[BUFF], key[BUFF], pt[BUFF], and ct[BUFF] as character arrays.

2.3: Declare len as 0 (to track valid characters).

2.4: Declare pi and ci as integers.

Step 3: Input:

3.1: Prompt "Enter the plaintext message" and store it in message.

3.2: Prompt "Enter the key (same length as the message)" and store it in key.

Step 4: Validate Key Length

4.1: If `strlen(key)` does not match `strlen(message)`, print an error and exit.

Step 5: Process Plaintext Message:

5.1: For each character in message,

 If the character is alphabetic, convert it to uppercase and store it in `pt[len]`.

Step 6: Increment len for each valid character.

6.1: Validate and Convert Key.

6.2: For each character in key (up to len),

 If the character is not alphabetic, print an error and exit.

6.3: Convert each character in key to uppercase.

Step 7 : Encrypt Message Using Vernam Cipher:

7.1: For each character in pt (up to len),

 Compute pi as the position of `pt[i]` in the alphabet (using ASCII values).

7.2: Compute k as the position of `key[i]` in the alphabet.

7.3: Calculate $ci = pi \wedge k$ to perform XOR between plaintext and key.

7.4: Convert ci back to a character and store it in `ct[i]`.

Step 8 : Output Encrypted Message,

8.1: Print "The ciphertext is:" followed by ct.

Step 9: Stop