Breaking RC4: A Cryptanalysis Assignment.

Consider an RC4 scheme with a word size of $n = 5$, where the internal state consists of a table S containing $2^n = 32$ words. We use a fixed-size key K which is 5 bytes long.

In this assignment, you will be provided with 2^{24} distinct ciphertexts, all generated from the same plaintext using 2^{24} different random keys, each 5 bytes in size, based on the RC4 scheme described above. The plaintext consists of a 6-digit passcode containing digits ranging from 0 to 9.

In this assignment, your task is to examine the vulnerabilities of the RC4 scheme and exploit them, using the provided list of ciphertexts, to retrieve the encrypted passcode.

Pseudocode for ciphertexts generation:

```
KeyScheduleAlgorithm($K$):
    For $i = 0$ to $2^{n-1}$
        $S[i] = i$

    $j = 0$
    $l$ = LENGTH($K$)

    For $i = 0$ to $2^{n-1}$
        $j = (j + S[i] + K[i \bmod l]) \bmod 2^n$
        SWAP($S[i]$, $S[j]$)

    RETURN $S$

PseudoRandomGeneration($S$, $m$):
    $i = 0$
    $j = 0$
    KeyStream[$m$]

    For 0 to $m - 1$
        $i = (i + 1) \bmod 2^n$
        $j = (j + S[i]) \bmod 2^n$
        SWAP($S[i]$, $S[j]$)
        KeyStream[$i$] = $S[(S[i] + S[j]) \bmod 2^n]$

    RETURN KeyStream
```

```
CiphertextGeneration(passcode):
    $m$ = LENGTH(passcode)
    NumberOfKeys = $2^{24}$
    Ciphertexts[ NumberOfKeys ]

    For $i = 0$ to NumberOfKeys - 1
        $K[5]$ = Random key of size 5 bytes
        $S[2^n]$ = KeyScheduleAlgorithm( $K$ )
        KeyStream[$m$] = PseudoRandomGeneration($S$, $m$)
        Ciphertexts[$i$] = passcode $\oplus$ KeyStream

    RETURN Ciphertexts
```

Provided materials:

- 4096 text files, each containing 4096 ciphertexts.

Deliverables:

1) Your program file should be named as Prog_Asgn_2_<Roll_No>.cpp or Prog_Asgn_2_<Roll_No>.py
2) Document your observations, provide a brief description of your approach, and include the cracked passcode in a PDF file.
3) Combine your program file and report into a zip archive file and name it as Prog_Asgn_2_<Roll_No>.zip. Upload this zip file.

References:

- https://www.usenix.org/system/files/conference/usenixsecurity13/sec13-paper_alfardan.pdf
- Section 3.9: https://toc.cryptobook.us/book.pdf.