МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

ЛАБОРАТОРНА РОБОТА №4

з дисципліни «Криптографія»

Варіант 4

Виконали:

Студенти групи ФБ-92

Шевченко Семен та

Щур Павло

Київ 2021

Завдання

Мета роботи:

Набуття навичок частотного аналізу на прикладі розкриття моноалфавітної підстановки; опанування прийомами роботи в модулярній арифметиці.

Завдання:

1. Написати функцію пошуку випадкового простого числа з заданого інтервалу або заданої довжини, використовуючи датчик випадкових чисел та тести перевірки на простоту. В якості датчика випадкових чисел використовуйте вбудований генератор псевдовипадкових чисел вашої мови програмування. В якості тесту перевірки на простоту рекомендовано використовувати тест Міллера-Рабіна із попередніми пробними діленнями. Тести необхідно реалізовувати власноруч, використання готових реалізацій тестів не дозволяється.

2. За допомогою цієї функції згенерувати дві пари простих чисел p, q і 1 1 p , q довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб pq ≤ p1q1 ; p і q – прості числа для побудови ключів абонента A, 1 p і q1 – абонента B.

3. Написати функцію генерації ключових пар для RSA. Після генерування функція повинна повертати та/або зберігати секретний ключ (d, p,q) та відкритий ключ (n,e) . За допомогою цієї функції побудувати схеми RSA для абонентів A і B – тобто, створити та зберегти для подальшого використання відкриті ключі (e,n) , (,) 1 n1 e та секретні d і d1 .

4. Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів A і B. Кожна з операцій (шифрування, розшифрування, створення цифрового підпису, перевірка цифрового підпису) повинна бути реалізована окремою процедурою, на вхід до якої повинні подаватись лише ті ключові дані, які необхідні для її виконання. За допомогою датчика випадкових чисел вибрати відкрите повідомлення M і знайти криптограму для абонентів A и B, перевірити правильність розшифрування. Скласти для A і B повідомлення з цифровим підписом і перевірити його.

5. За допомогою раніше написаних на попередніх етапах програм організувати роботу протоколу конфіденційного розсилання ключів з підтвердженням справжності по відкритому каналу за допомогою алгоритму RSA. Протоколи роботи кожного учасника (відправника та приймаючого) повинні бути реалізовані у вигляді окремих процедур, на вхід до яких повинні подаватись лише ті ключові дані, які необхідні для виконання. Перевірити роботу програм для випадково обраного ключа $0 < k < n$. Кожна з наведених операцій повинна бути реалізована у вигляді окремої процедури,

інтерфейс якої повинен приймати лише ті дані, які необхідні для її роботи; наприклад, функція Encrypt(), яка шифрує повідомлення для абонента, повинна приймати на вхід повідомлення та відкритий ключ адресата (і тільки його), повертаючи в якості результату шифротекст. Відповідно, програмний код повинен містити сім високорівневих процедур: GenerateKeyPair(), Encrypt(), Decrypt(), Sign(), Verify(), SendKey(), ReceiveKey().

Хід роботи

Для виконання поставлених завдань було реалізовану функцію генерації чисел заданої довжини та функцію перевірки отриманого числа на простоту. За допомогою цих функцій було отримано дві пари ключів p і q. Було реалізовано функцію отримання секретного та публічного ключів. Далі було реалізовано функції шифрування і дешифрування , отримання цифрового підпису а також його верифікації. Після цього було створено два абоненти та функції отримання та надсилання ключа.

Результати:

```
p =  833673949245969508129335552824241648678543423797363942322022493518501610266603
q =  943962980502372390819568626585137077454746488066701865513438754237132093183 51
p1 =  1156855358863984754170610580695412613042911002228157445424762569464376154043 17
q1 =  609325070045422016822878892864482916051490117825130453884910444794828095509 99
```

```
A public key:
e = 189581176645749517927380605365729699187284565930069880200069903622595528057965932692053545882770016214668261805297094144728233759446282752801614298513338 71
n = 786957345897408904929591227133303797738395061322686565123478290871779521090593779705029171951184962883913450030567876716066208045696205784542112007091653
A private key:
d = 293785218072775995898925494477387202453345890187154891822361012281446210632349860780056041436182370198129199362753881626167360581228198166760093253064483 1
p = 833673949245969508129335552824241648678543423797363942322022493518501610266603
q = 943962980502372390819568626585137077454746488066701865513438754237132093183 51
```

```
Plain message: 116018354809793759245452081638310668425510420216980512433057194560535107124095097020869066017675957638559329381688000912077614497535 41

Encrypted message: 693520133459802637499045800155509622970867181564818758152876541690419037322821737330774448161628466155895066479319533762530160550638867380067860237068227 20

Signature for encrypted message: 411322945147327795733769942242161148369787334360470725584855594688024324547961794751639828231433290633169435921488281774883116470213308449553203513016549 8

Verified message!

Decrypted message: 116018354809793759245452081638310668425510420216980512433057194560535107124095097020869066017675957638559329381688000912077614497535 41
```

Перевірка роботи функцій була реалізована за допомогою сайту https://www.dcode.fr/rsa-cipher: