**Лабораторна робота №4**

**Вивчення криптосистеми RSA та алгоритму електронного**

**підпису; ознайомлення з методами генерації параметрів для**

**асиметричних криптосистем**

**Виконали:**

Анучін Максим ФБ-11

Ступак Ярослав ФБ-11

**Мета роботи**

Ознайомлення з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практичне ознайомлення з системою захисту інформації на основі криптосхеми RSA, організація з використанням цієї системи засекреченого зв'язку й електронного підпису, вивчення протоколу розсилання ключів.

**Порядок виконання**

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  p1 , q1 довжини щонайменше 256 біт. При цьому пари чисел беруться так, щобpq < p1q1; p і q –прості числа для побудови ключів абонента А, p1 іq1 – 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 А і B – тобто, створити та зберегти для подальшого використання відкриті ключі (e, n), (e1, n1) та секретні d і d1.

4. Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів А 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 А и B, перевірити правильність розшифрування. Скласти для А 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.

**Хід роботи**

Написав функцію яка приймає 2 аргументи: мінімальну і максимальну величину простого числа яке буде згенеровано. У випадку якщо аргумент наданий тільки один, він сприймається як мінімальна довжина в бітах

Далі генеруємо дві пари простих чисел для абонентів A B мінімальною довжиною в 256 біт:

ap = 139672102951664142186453343489092249579032609880927332775446586938425219739863

aq = 159691178035788586299454291475781380222731465646510389662936615379262121803767

bp = 129380935509241031518642830120924231012205656097411878296711164423826926040303

bq = 123462647380084836374109519510234993652684361136623104776725248243497930500851

Після чого генеруємо пари з відритих і закритих ключів. В якості випадкового числа e використовуємо значення 2\*\*16+1

an = 22304402659087191036226126687890863893914426453739417042454916968319531781737168297779266383365877953479989676959689630439144309083379865913733224073463921

ad =

19975163728425233127686956279542525207524290581645903908827173072029209131719212786250629190776209642595511555027558608732162977108926842009568825204183861

bn = 15973712818482922415261167220681689841133017220531258195398135663137673523041961249736304393686351825014698943990712551959302200121588252059016807001797853

bd = 1730280411041247939727772496446576989825644891413268869938071090721490827167448738503182836663918717012516840930377837742446891135058313262331735870564373

Далі шифруємо повідомлення A «885555555588» і B «606060606060»

Після шифрування отримуємо наступні числа  
Для А:

7909653324632566127628535593417023480904067126093136227783271843865137077158472766456055888166058638434909851549543048258559705114473047560364334304050690

Для В:

11127662413153133216318610504343927585627678717194297316567088618666950431895956075112462483240487013727488449837333935764747630199867269606345144728422709

При дешифруванні отримуємо назад наші повідомлення  


Далі перевіряємо функції підпису і верифікації, повідомлення залишаємо тими самими:

Підписане повідомлення А:

885555555588, 20876033750325729773755035699275888745589032367932960251686308284777287485559244609637151802019399145173964849258600445795796606491219897982193266316378346

Підписане повідомлення В:

606060606060, 7732985231191521683823214866490784445037280337396379192601097424119264662183986319832890917598248724894264170170387950626824348017399914389686631204952278

Далі для перевірки верифікуємо повідомлення А з правильним ключем, і повідомлення В з неправильним:



**Висновок:** Під час виконання лабораторної роботи ми ознайомились з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практично ознайомились з системою захисту інформації на основі криптосхеми RSA, організували з використанням цієї системи схеми засекреченого зв'язку й електронного підпису, вивчили протокол розсилання ключів і закріпили навички набуті у попередніх лабораторних роботах.