

Міністерство освіти і науки України  
Національний технічний університет України  
“Київський політехнічний інститут ім. Ігоря Сікорського”  
Фізико-технічний інститут

Лабораторна робота № 4  
з предмету «Криптографія»  
«Вивчення криптосистеми RSA та алгоритму електронного підпису; ознайомлення з  
методами генерації параметрів для асиметричних криптосистем»  
Варіант 6

Виконали:  
Студенти 3 курсу,  
ФТІ, групи ФБ-92  
Друзь Данило  
Чикрій Кирило

**Мета:** 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.

### **Постановка задачі:**

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  $p_1, q_1$  довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб  $pq \leq p_1q_1$ ;  $p$  і  $q$  – прості числа для побудови ключів абонента А,  $p_1$  і  $q_1$  – абонента В.

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 В – тобто, створити та зберегти для подальшого використання відкриті ключі  $(e, n)$ ,  $(e_1, n_1)$  та секретні  $d$  і  $d_1$

4. Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів А і В. 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 і В, перевірити правильність розшифрування. Скласти для А і В 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:**

В ході виконання роботи, були створені функції знаходження простого числа в діапазоні, шифрування, дешифрування, надсилання, та отримання повідомлення, а також був описаний процес автентифікації співбесідника. Основні для цього функції це функція знаходження простих чисел (включаючи і тест Міллера-Рабіна) та функції зведення числа в степінь за модулем (реалізована за схемою Горнера).

Єдиною проблемою в даній лабораторній роботі була функція, що тестувала кандидатів на просте число. Вона зайняла найбільший час написання, все інше було набагато легше.

### **Результати**

Data:

Sender Public Key: e =

81392535834141116297852232238634724587383319788511081374589664935585181510033  
75378034994530887706254248284166434260722621128963360780563038603084227552327  
n =

10240602304677163484526559680401311268278782060542938141061474841333768953693  
45548969952323352236117978724848844177224278324138583606869557586468768337207  
9

Sender Secret Key: d =

80304202595318060738750365235248162271704068433885103367354634406743068849946  
33061898207904919211109732061675596482434745271675897827854379273141457354031

p =

10131802291443625226199667148974734989677505953526759960717174567287767920117

3 q =

10107384658823653316756959211453967730384634878970367309519158545741029498872  
3

Data:

Receiver Public Key: e =

95804082077566947784419625767153434737091153397676382980764391453603061393902  
29986790765539920091618943683695432181394831947048506186676631093113010654613

n =

11702773790462097053440912658412589984891087495292025391479303432611383876433  
54512401826656409479453818551670426029339186169374057043894261663063575808911  
3

Receiver Secret Key: d =

11332089722267423046353171476732100302254630708948775530033795987392539822703  
60270390396420610252757804742810406366554546632488703771404119823753902656585

7 p =

10532299374388797139118769576680332841098978641038372348659801815998933197627

1 q =

11111318976480597854350107006191931072006564131918342611909911202615649204930  
3

Generated message:

47715263362300779509979828465353559915847896835295987107314004628996813789287  
3614467199664144780595635502337730095116237345744156885475271376317563687731

Authentification...

Authentification success

Encrypted:

89083712365042729156979954286018978915139363678937331110192156862289951604661  
30693682490581052451360920191929102281944477871240889403982220285210619176908

Signed:

29444243344335920954402858047143587314604805629896193466678683920022916800230  
10281084280393438091428631692086305227173077696987772706805922913610076820341

Checking...

Look`s like original

Got message:

47715263362300779509979828465353559915847896835295987107314004628996813789287  
3614467199664144780595635502337730095116237345744156885475271376317563687731

#### Висновки

Під час даного лабораторного практикуму ми ознайомилися з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA; практично ознайомилися з системою захисту інформації на основі криптосхеми RSA, організація з використанням цієї системи засекреченого зв'язку й електронного підпису, вивчення протоколу розсилання ключів.