

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ**  
**УКРАЇНИ**  
**“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ**  
**СІКОРСЬКОГО”**

**КРИПТОГРАФІЯ**  
**КОМП'ЮТЕРНИЙ ПРАКТИКУМ №4**

**Вивчення криптосистеми RSA та алгоритму електронного  
підпису; ознайомлення з методами генерації параметрів для  
асиметричних криптосистем**

*Виконали роботу:*

*студент ФБ-23 Хоменко Гліб*

*студент ФБ-23 Ткачук Андрій*

*Варіант 3*

**Київ 2024**

## Мета та основні завдання 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

## Постановка задачі

0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.

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  $pq \nmid p_1q_1$ ;  $p$  і  $q$  – прості числа для побудови ключів абонента  $A$ ,  $p_1$  і  $q_1$  – 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  $(e, n)$ ,  $(e_1, n_1)$  та секретні  $d$  і  $d_1$ .

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  $A$  і  $B$ , 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  $0 < k < n$ .

# Хід роботи

## *1. Генерація випадкових чисел з перевіркою на простоту тестом Міллера-Рабіна*

Приклад згенерованих не простих чисел

```
1 14070509468937164028239004270977789194037530802969821017878814657791945066015
2 35712154072248969624306423075685397913424166238387870488311272108578379865989
3 114805636892394077031781886428945017317115203787691523353462611875727873412455
4 104141203189334536797476421287073971830853574141677707221776666569795653837063
5 66225825529170364214126116128298540357996987086976149200812863256328330861885
6 62869695033744434476417510031060712443835130445608241017503221609834540040831
7 45193033943104073890941063853065112170315401214421272973221616352794162094117
8 11531611782124417726367745302661169955525607655468373188995768013846013131085
9 47019449712599060770497034353029855340017235903241676757420689291648536288801
10 20555219089392987257921952133449263178351226312950740056768263585560312426227
11 35519492793571605321308484993358125263755347343850973872427646919278499456615
12 29744489194231981425719450684855969376787995257924477988600736453888476588101
13 111143438774771913743131721725352731739158494699789809765455624811925771720355
14 100487476318066678787546321113333872599067670519072537832077312018441000668417
15 73462975921253776394387637967360379012759602854887263232700557517748703015169
16 65883198687776734654774353745175794830123123858401160534120122270420615980035
17 70836061293898453429358734845816483384923651273489485742201337706453127121483
18 14886159054081761777961010985598297522347640299995663007768737935511291800735
19 109437902943531830184866885382752780735817829471542211053146085321873594330655
20 36642646094795499842277856442573312670422638044201400278017212947362877340881
21 101281981843198291420789205957919626079371483669330562569274839877055042216401
22 4118965504295178242017077744654270766601420832915259296492695690190792632959
23 31463008145600597622049262476924386463417386714988348814227109176126662784467
24 92639982165346271611522943777898629627094765614575010408703164275298317644745
25 93246961047954151451586597426929186559308610866338248311590782590853754705875
26 8563048523813579141370189279390331884406672006655645505297920995034293941749
27 93343255455456489992996172985760844480623424715084020480207791891792360310069
28 44584105257976724575404404972758806945568361532101751201811053289143932790905
29 52578650881050806976646617360663188356822424742070534887712961476828616541761
30 24973007573631976228125721326686034450677428255768963333078511014804345749541
31 26360025370991476919784153844221024617892892783720362239945913657538882454205
32 16381505835431610211458940428719019377522833181964689535929251591769581688327
33 29031592674305911981680687254569928102083978848139747233317725662342542755897
34 83417987616141171926427682584323399489519243435690709968236313606357045909761
35 85849212244979450731148752789781434714933189468055612459345418816715339509943
36 111190003723631037956456250654772649266751571556795510400173475299772433501175
37 5366889800284804085589189888275192283490900651458398800022813560183278359719
38 99799345477635930066459307714846215843138255001096292087708543320945778371009
39 9076130325293396512586144732533850712454102212821745532576353310465794045403
40 77335508639540703169130034959676238951478247150044604555593190023103303301881
41 2667578147321678733924238514732309522004141721469986748318961652657085749681
```

2. Генерація  $(p, q)$  та  $(p1, q1)$  довжини щонайменше 256 біт для користувачів A і B відповідно

$p=18237052018850384751518250401777005005759020965879163026290152325594413407$   
319

$q=32526987514378734599428393420423246762281375026882375641346930385020386435$   
781

$p1=9154919973891075751310805151750023669982438254729537948790736976888043174$   
2131

$q1=9680294004524744463796491183454963674933421356979186516290642870067300703$   
7877

3. Генерація пар відкритого ключа  $(e, n)$  та секретного ключа  $(d, p, q)$

A public key	e	$2^{16} + 1 = 65537$
	n	593196363316221960037 844197364380098454291 968720920100574265510 082183786638594156036 281446177371749260689 467107572079268864705 718787780799395674188 881139
A private key	d	257754441584998592489 703361587278210227548 902044122277553952102 333191139235940006010 214061936940329830790 685540270864015483991 553126956263180194946 696313
	p	182370520188503847515 182504017770050057590 209658791630262901523 25594413407319
	q	325269875143787345994 283934204232467622813 750268823756413469303 85020386435781
B public key	e1	$2^{16} + 1 = 65537$

	n1	886223169351616107173 230418609519623081887 942241946064499400471 095674488205405661209 757666763973975731226 158285143219692260226 080919289502106047101 3695887
B private key	d1	5229956711146849041157 187492583087560052351 809534996412801747504 477822681207797256849 223844469661106792780 034140355402144058174 356199239107144457758 631713
	p1	915491997389107575131 080515175002366998243 825472953794879073697 68880431742131
	q1	968029400452474446379 649118345496367493342 135697918651629064287 00673007037877

#### *4. Шифрування, розшифрування, створення повідомлень з цифровим підписом для абонентів A,B*

Повідомлення генерується випадковим чином довжиною 256 біт

Зашифрування (encrypt\_RSA\_message) відбувається згідно формули  $C = M^e \bmod n$

Розшифрування (decrypt\_RSA\_message) відбувається згідно формули  $M = C^d \bmod n$

Цифровий підпис створюється згідно формули  $S = M^d \bmod n$

Перевірка підпису згідно формули  $M = S^e \bmod n$

Оригінальне повідомлення	М	1046531486300246209086 5777769781812108820564 7423862821678288211741 772485209906
А	зашифрування	3235263424633648350419 7815618597316725633812 5612464652092977947660 3914768089647981904244 7509604518701806754882 1609715887858465188420 806233302807955536031
	розшифрування	1046531486300246209086 5777769781812108820564 7423862821678288211741 772485209906
	Цифровий підпис	1556358328700563991518 6238274499562080915448 2953504098284062103203 8982904435032912030770 0634434231618864135587 4877594410446547430687 945767511945941707611
В	зашифрування	3470375061654699735762 0349366546111711739043 2766063510354192756905 9596884422445801756522 0805828291517570017603 3217958279354795835911 0150339048908458427076
	розшифрування	1046531486300246209086 5777769781812108820564 7423862821678288211741 772485209906
	Цифровий підпис	2712379685872900504812 1435928698717475180120 9077304428388878462127 5612860632297387916858 1226166360145336274919 5976710846046590237603 577786894470299977202

Перевіримо за допомогою <https://www.dcode.fr/rsa-cipher> для А:



### Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:  
e.g. type 'caesar' 

★ BROWSE THE [FULL DCODE TOOLS' LIST](#)

### Results

  Decryption using C,D,N

10465314863002462090865777697818121088205647  
423862821678288211741772485209906

RSA Cipher - [dCode](#)

Tag(s) : Modern Cryptography, Arithmetics

### Share

### dCode and more

dCode is free and its tools are a valuable help in games, maths, geocaching, puzzles and problems to solve every day!  
A suggestion ? a feedback ? a bug ? an idea ? [Write to dCode!](#)

## RSA CIPHER

Cryptography > Modern Cryptography > RSA Cipher

### RSA DECODER

Indicate known numbers, leave remaining cells empty.

★ VALUE OF THE CIPHER MESSAGE (INTEGER) C=  
3235263424633648350419781561859731672563381256124... 

★ PUBLIC KEY E (USUALLY E=65537) E=  
65537 

★ PUBLIC KEY VALUE (INTEGER) N=  
5931963633162219600378441973643800984542919687209... 

★ PRIVATE KEY VALUE (INTEGER) D=  
2577544415849985924897033615872782102275489020441... 

★ FACTOR 1 (PRIME NUMBER) P=  
1823705201885038475151825040177700500575902096587... 

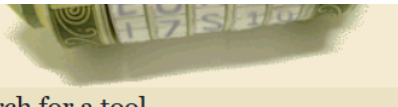
★ FACTOR 2 (PRIME NUMBER) Q=  
3252698751437873459942839342042324676228137502688... 

★ INTERMEDIATE VALUE PHI (INTEGER) Φ=


★ DISPLAY ☐ PLAINTEXT AS CHARACTER STRING  
☐ COMPUTED VALUES (C,D,E,N,P,Q,...)  
☒ PLAINTEXT AS INTEGER NUMBER  
☐ PLAINTEXT AS HEXADECIMAL FORMAT

 CALCULATE/DECRYPT

Для В:





### Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:  
e.g. type 'random' 

★ BROWSE THE [FULL DCODE TOOLS' LIST](#)

### Results






  Decryption using C,D,N

10465314863002462090865777697818121088205647  
423862821678288211741772485209906

RSA Cipher - [dCode](#)

Tag(s) : Modern Cryptography, Arithmetics

### Share

### dCode and more


dCode is free and its tools are a valuable help in games, maths, geocaching, puzzles and problems to solve every day!  
A suggestion ? a feedback ? a bug ? an idea ? [Write to dCode!](#)


## RSA CIPHER


Cryptography > Modern Cryptography > RSA Cipher


### RSA DECODER

Indicate known numbers, leave remaining cells empty.

★ VALUE OF THE CIPHER MESSAGE (INTEGER) C=  
3470375061654699735762034936654611171173904327660... 

★ PUBLIC KEY E (USUALLY E=65537) E=  
65537 

★ PUBLIC KEY VALUE (INTEGER) N=  
8862231693516161071732304186095196230818879422419... 


★ PRIVATE KEY VALUE (INTEGER) D=  
5229956711146849041157187492583087560052351809534... 

★ FACTOR 1 (PRIME NUMBER) P=

★ FACTOR 2 (PRIME NUMBER) Q=

★ INTERMEDIATE VALUE PHI (INTEGER) Φ=

★ DISPLAY ☐ PLAINTEXT AS CHARACTER STRING  
☐ COMPUTED VALUES (C,D,E,N,P,Q,...)  
☒ PLAINTEXT AS INTEGER NUMBER  
☐ PLAINTEXT AS HEXADECIMAL FORMAT

 CALCULATE/DECRYPT

## 5. Протокол конфіденційного розсилання ключів

k генерується випадковим чином  $0 < k < n_1$

1. Абонент А формує повідомлення  $(k_1, S_1)$ 
  - a.  $k_1 = k^{e_1} \bmod n_1$
  - b.  $S = k^d \bmod n$
  - c.  $S_1 = S^{e_1} \bmod n_1$
2. Абонент В знаходить k, S за допомогою секретного d:
  - a.  $k = k_1^{d_1} \bmod n_1$
  - b.  $S = S_1^{d_1} \bmod n_1$
3. В перевіряє підпис А за допомогою відкритого ключа e абонента А:
  - a.  $k = S^e \bmod n$

A generate secret k

4443513688928164754647087668056319523838403899146968078356057775723473952481  
15332252281472220746959773866349839871256792930486690808033120721354744097094

A formed message  $(k_1, s_1)$ :

$k_1 = 3920073274341323080792740515701606690813183015108501164513758380843864914$   
 $468764165212222167032731483273510918477629105712485956035465121780254156969134$   
574

$s_1 = 8726855171203952123165828936619256741894851047291441733004294835369092310$   
 $669990558480003660263421525030309822443410800955295318527662865910763749766580$   
852

B received  $(k_1, s_1)$  and formed message  $(k, s)$ :

$k = 44435136889281647546470876680563195238384038991469680783560577757234739524$   
 $811533225228147222074695977386634983987125679293048669080803312072135474409709$   
4

$s = 46935681523791527694063179730725628931976720859268082311921171508113801434$   
 $733425390655046037423424204145561903906485095766680264680758982858170954410018$   
5

Verification state: True



## **Висновки**

У ході виконання лабораторної роботи, реалізували перевірку числа на простоту за тестом Міллера-Рабіна. Також реалізували криптосистему RSA з цифровим підписом. Автоматизували перевірку змістовності розшифрованого тексту.