МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ

КРИПТОГРАФІЯ

КОМП'ЮТЕРНИЙ ПРАКТИКУМ №4

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

Виконали:

Студенти групи 3 курсу
груп ФБ-96 та ФБ-94
Ігнатенко Артем
Васюченко Георгій

Київ 2021

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 p1,q1 довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб p1,q1; p,q – прості числа для побудови ключів абонента A, p1,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) , (e1,n1) 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.

Хід роботи:
Тест простоти Міллера-Рабіна:

```
def MRabintest(n, k):
   n = int(n)
   if n != int(n):
       return False
```

```
    check = [2, 3, 5, 7, 11, 13, 17, 19]
    for i in check:
        if n % i == 0:
            return False
        s = 0
        d = n-1
        while d % 2 == 0:
            d = int(d//2)
            s += 1
        for i in range(0, k):
            a = random.randint(2, n-1)
            x = pow(a, d, n)
            if x == 1 or x == n - 1:
                continue
            for r in range(0, s-1):
                x = pow(x, 2, n)
                if x == 1:
                    return True
                if x == n-1:
                    break
            else:
                return False
        return True
    else:
        return False
```

```
"D:\Python stuff\Labs_Crypta\Labs\Scripts\python.exe" "D:/Python stuff/Labs_Crypta/lab4.py"
RabinTEST 6: False
RabinTEST 10: False
RabinTEST 13: True
RabinTEST 19: True
```

Згенеровані p, q, p1, q1:

```
p: 81183483224185091730255907243695018023198224220517123574997519800715927965859
q: 26585576238398335836823739632734204221722395230968918128728515070048890391371
p1: 52668287807993441348021169159080283556620490434134315293931259428236514088923
q1: 10721990355135270615189205807763407328209284203959559457222932052315559260602 3
check p*q < p1*q1: True
```

Кандидати p та q що ‘не пройшли’ перевірку на простоту:

```
32456929263400872679646171588623567473716537703838695412114641881464410952551
False
10193232888632821482528861308724249868159919840168007118858184928844819561455
1
False
21463522439117480633484414491430654016229747832088690049091030319906830975480
False
54129108667118349056418823408635828780952024389332911325460957026854148329232
False
11003323824297711360319663397334173620599236554378608913902678354114560073959
9
False
27845871965979174958582901448044929216690633789417310579912167555367946559631
False
```

10823673899004752900384949648613428250697718816295039417829440514199094902441
6
False
46627330402288128264945578301290176500596744757657991679885462270182153876432
False
26219587850886419992371916933262673739744127883778444078743545004756920048366
False
60955376488063874356450865281005209046851801081942784160134942955871522233552
False
75596678032289947617649237153183723363150749424629603620480235867467661786722
False
72824060234939530463724303937311588202820276046215756612971672007460556800374
False
96735718477823355887555738885120382382180526335270831845987983023064779039745
False
93067470469454119631017101514378299233175050461204762318602704301225561472065
False
32228768854444990631371625953350704388833337740204404028822201157855580541920
False
28017919830314438192880276021452081309287994506605246868541190164997415549014
False
97776753122943363463236288056049357659094150646930332873542012351796280300693
False
27328084486374065099691762783558191157141886015481806071886162095421582322809
False
36517391161584873466675835906094492013994350682179353778969362641793070008 89
False
99822894123547142537266909850827206256995587098447985661597369046375501506008
False
96638981957809170695424024438412653776974765869636196745845089897184498810375
False
45374300928616244691374450920336186810659883658320893112434220012909932718841
False
53407849915660518490595176193826407455163143068669597744185976517520758754299
False
46951351431993645573818485571921583835945665204526660222537274717499348237504
False
30502089424393533176097822657512699952740044722252181155387261814332209575444
False
85639612172401260161174170674378817182099313990177484672805722710580667222493
False
59821233047323244264859876891141496785097991495690933697912032880790721923710
False
11153917758377359119438545202086645829391485978496910188802580781809698516050
False
59185359030969317999763633262545671824153105514735618296685910827217808346814
False
52088247293865327932803076766989818152289419680702803919181619942330048898574
False
65351616994759316729220360416393906894714480060928185332502090016238025809089
False
19847193186878800855614226078650489524615149482846490123364939215278187225979
False

```
1038800463432085141773402139648538692889705046787956337317913626394696419529 6
3
False
484972572128791962174011517206791652793536215998099761790820938714189023349 75
False
383408161792566117988778953373248665584600906788611624867603425512838020223 48
False
588634248373940002105419766397761074806471702445904859433232302031516614839 36
False
100026242577014408004277241516677835988865579893869327790078388374763489573 546
False
562334553511246396174934866326194679551250871859610392319071818178432838215 59
False
336694675602789891949400017258799313159016855753598524975779115795645259285 9
False
829713591868464006154962685547584669602312439182135877214793514277483599786 82
False
773202203494288878652648611385908228967753827782750290628587414979575159403 77
False
```

Чисельні значення прикладів ВТ, ШТ, цифрового підпису для А і В:

```
M: 242535364960262012300259158993
C: 606153455996618931097209494685231389541850289673571832764243387739692054295170743925959790211741864550589345776024279155843913937486272920902861095549882
M_d: 242535364960262012300259158993
check M == M_d: True

A public:
e: 108055502967968079737873565560243758009964769571200556183486370584658966842581459688745082756332270130415101467887942234555463254628930965365391996935034 9
n: 215830968255530509234411354986539217686146934238105187315361636909101097329552079515445971733399281973565541078504301640406669991752605063937429043620268 9
d: 561600285113652093927672178800062210901564944574243577216513950898487962079800324275341942453380125202175172099972397698915259576800241064796882856109629

B public:
e1: 994672428278795467376342272070863343034075231881954219509960447662338350931810561616434280408689267612486216077225432110550380990281233093953533184847715
n1: 564708873898794241729490594724822530547735221573890065745622828084774219034877959524661229609890734025215569667103384584628645239558649904502968512738322 9
d1: 154969439004525969041721019341791236169097057428532763897738709947747353965761908972569286594651979145192494340326344247998333464352331889842670574640330 7

check n1 > n: True
k:  655638682781165880497705017366561945357897981913273872336269297464366580507617262649038541809913701811209103744675156923570643093877567457785996759283760
k1:  315708923558204659394357741943475355214562502995974883028378814609700467909137816413112084488088057314940481439965329954746490404841872740908596340729968 92
S1:  269880841605418266418159061570946475528310953746027987368048978559143420992180404724566119351299548834809753741939752591835186977339700448341854184660917 60
S:  614261305730357086139816583417680382520351090484066812039927651804288967571681564944313360225580525299218020626991865878440499402263641599546487861558968
k_r:  655638682781165880497705017366561945357897981913273872336269297464366580507617262649038541809913701811209103744675156923570643093877567457785996759283760
S_r:  614261305730357086139816583417680382520351090484066812039927651804288967571681564944313360225580525299218020626991865878440499402263641599546487861558968
Verify: True
```

Розсилання ключів:

```
A public:
e: 108055502967968079737873565560243758009964769571200556183486370584658966842581459688745082756332270130415101467887942234555463254628930965365391996935034 9
n: 215830968255530509234411354986539217686146934238105187315361636909101097329552079515445971733399281973565541078504301640406669991752605063937429043620268 9
d: 561600285113652093927672178800062210901564944574243577216513950898487962079800324275341942453380125202175172099972397698915259576800241064796882856109629

B public:
e1: 994672428278795467376342272070863343034075231881954219509960447662338350931810561616434280408689267612486216077225432110550380990281233093953533184847715
n1: 564708873898794241729490594724822530547735221573890065745622828084774219034877959524661229609890734025215569667103384584628645239558649904502968512738322 9
d1: 154969439004525969041721019341791236169097057428532763897738709947747353965761908972569286594651979145192494340326344247998333464352331889842670574640330 7
```

Перевірка результатів на сайті:

Get server key

Clear

Key size 256

Get key

Modulus 82321E2ADD8A79C826C101220A6B8CC92549A3D70E6EB5078E486AD5EC61F37B

Public exponent 10001

Encryption

Clear

Modulus 82321E2ADD8A79C826C101220A6B8CC92549A3D70E6EB5078E486AD5EC61F37B

Public exponent 10001

Message i hate this site Text

Encrypt

Ciphertext 67AF986216E3104EC9C8D5303C9E2BAC15E65CE94FC810B9D31881068CD58342

```python
#code for testing the site
m = 'i hate this site'
m = int(m.encode().hex(), 16)
n = '82321E2ADD8A79C826C101220A6B8CC92549A3D70E6EB5078E486AD5EC61F37B'
e = '10001'
e = int(e, 16)
print(m)
n = int(n, 16)

c = Encrypt(n, e, m)
print(c)
c = hex(c)[2:]
print(c)

```

Run: lab4

```
139737210148642340706508095433254466661
46898473350342419812494477193674778081081772828862619628819284149914226230082
67af986216e3104ec9c8d5303c9e2bac15e65ce94fc810b9d31881068cd58342

Process finished with exit code 0
```

Sign

Clear

Message i hate this site — Text

Sign

Signature 018E8FCD831DC1C79421A30B25ED85E141C8310BA2E8F9A4FF661C2FE5F683FC

Verify

Clear

Message i hate this site — Text

Signature 018E8FCD831DC1C79421A30B25ED85E141C8310BA2E8F9A4FF661C2FE5F683FC

Modulus 82321E2ADD8A79C826C101220A6B8CC92549A3D70E6EB5078E486AD5EC61F37B

Public exponent 10001

Verify

Verification true ✔