НТУУ "КПІ ім Ігоря Сікорського"

Фізико-технічний інститут

КРИПТОГРАФІЯ КОМП’ЮТЕРНИЙ ПРАКТИКУМ №4

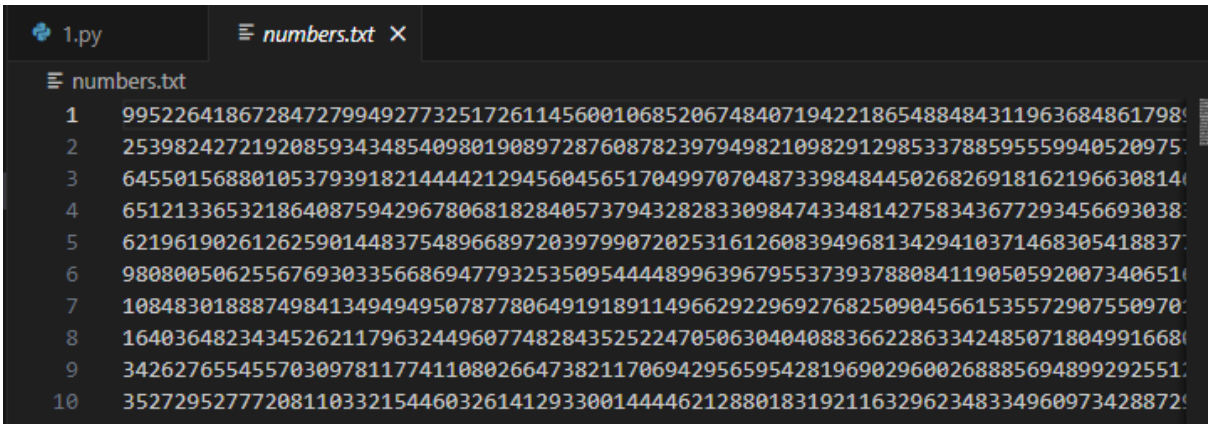
Вивчення системи RSA та електронного підпису

Виконав:

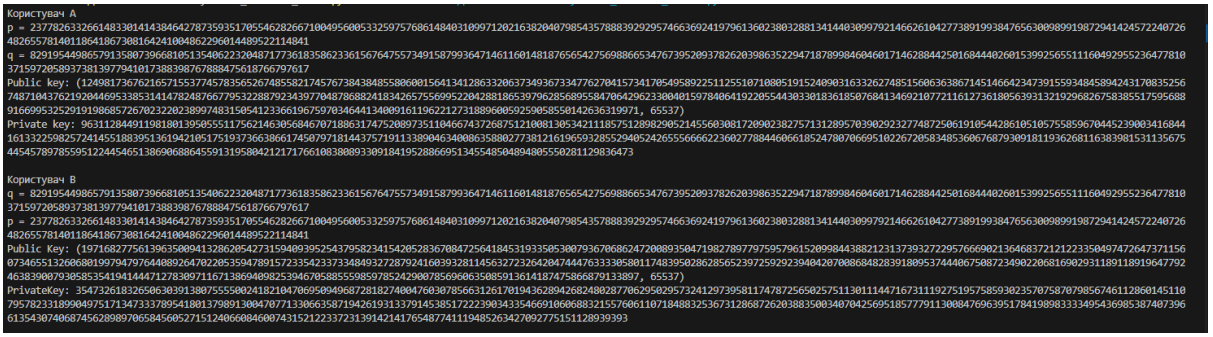
студент групи ФБ-14 Хаща Іван

Київ 2023

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.

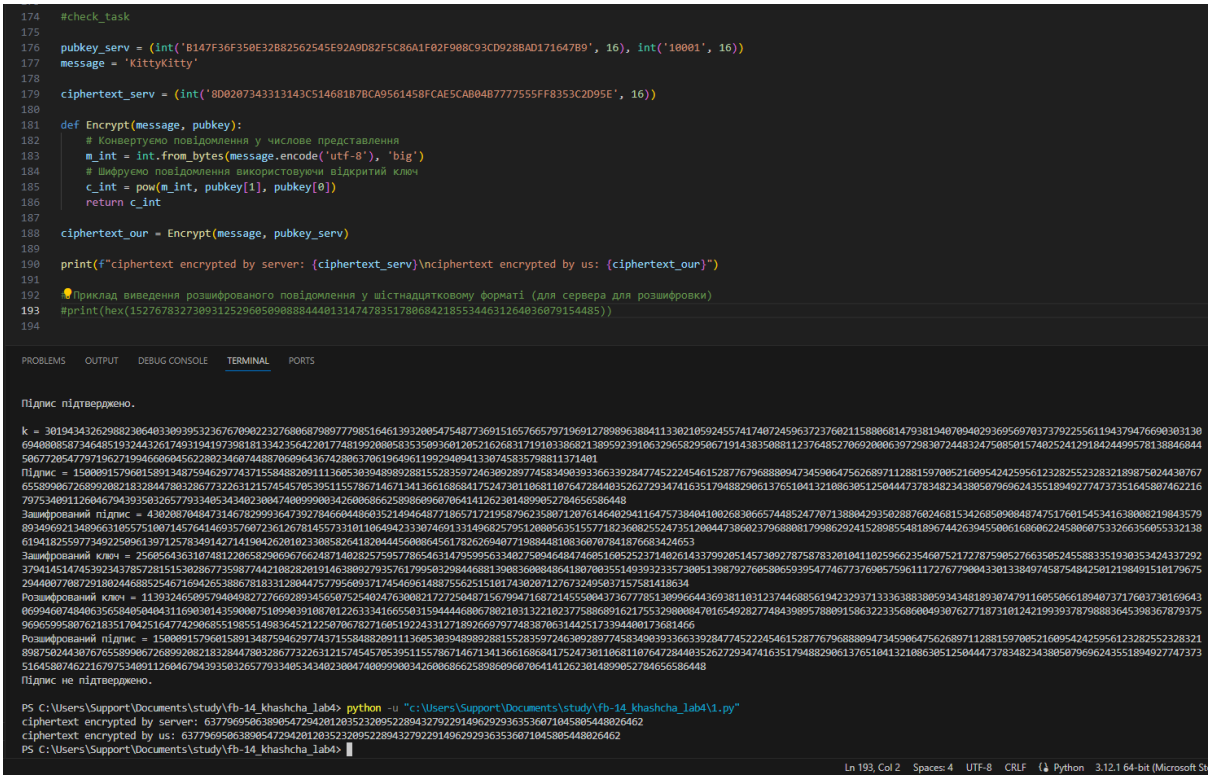


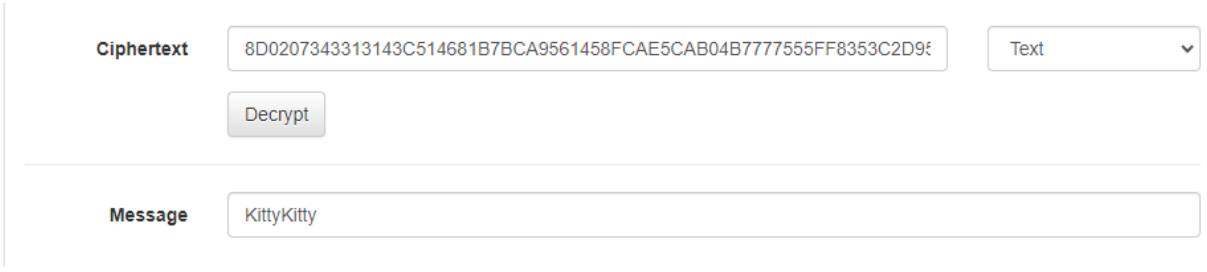
За допомогою цієї функції згенерувати дві пари простих чисел p, q і 1 1 p , q довжини щонайменше 256 біт. При цьому пари чисел беруться так, щоб pq p1q1 ; p і q – прості числа для побудови ключів абонента А, 1 p і q1 – абонента В.

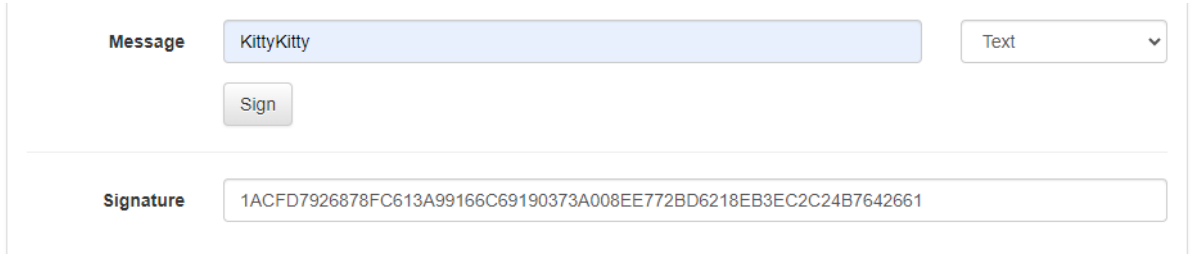


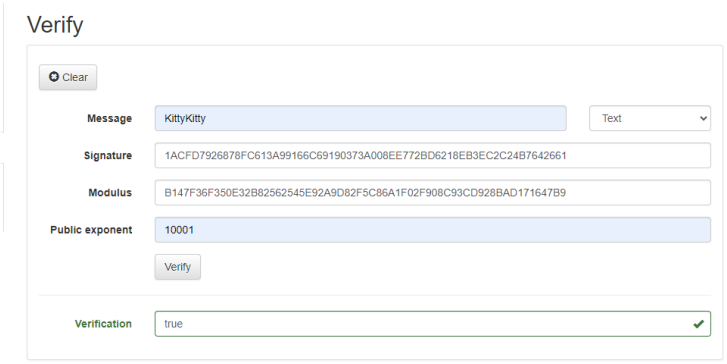
Написати програму шифрування, розшифрування і створення повідомлення з цифровим підписом для абонентів А і В. Кожна з операцій (шифрування, розшифрування, створення цифрового підпису, перевірка цифрового підпису) повинна бути реалізована окремою процедурою, на вхід до якої повинні подаватись лише ті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


Повідомлення: 822745593453514104214033352515217450065982645233704446784555818551768085808310037364973848574969595859828157808451279655862683974560215178730630155991855960959169226677713165184290826862376056807498020647381219237166620675933715738190984336775094502445732011602194692859297748187588516783061189381554558055133976804287702849509302472456277157768329112101678119811741653693747642665110683071290032377017230637872548050513149380284237945835918775705402206362318726395792060727164690207628775840715375203606964965
Зашифроване повідомлення за допомогою відкритого ключа A: 215575298608922637714325616771814469289536794982693462420884203529372454932815203404574536108190812125812681866018831431208470853546401936857133932217944574137305048890033199235869296103641894121574413287022444195229559449817283072582063802553013967694284913332538957262153990886330776020357333962936446303568966886447011310826065422059487505595232382684697409209869606111162875491700573527949766674880352684102079266281994116394240639045283316240350104316599979900826669176076424857884127683305079127886593498
Розшифроване повідомлення за допомогою секретного ключа A: 822745593453514104214033352515217450065982645233704446784555818551768085808310037364973848574969595859828157808451279655862683974560215178730630155991855960959169226677713165184290826862376056807498020647381219237166620675933715738190984336775094502445732011602194692859297748187588516783061189381554558055133976804287702849509302472456277157768329112101678119811741653693747642665110683071290032377017230637872548050513149380284237945835918775705402206362318726395792060727164690207628775840715375203606964965
Повідомлення з цифровим підписом: (822745593453514104214033352515217450065982645233704446784555818551768085808310037364973848574969595859828157808451279655862683974560215178730630155991855960959169226677713165184290826862376056807498020647381219237166620675933715738190984336775094502445732011602194692859297748187588516783061189381554558055133976804287702849509302472456277157768329112101678119811741653693747642665110683071290032377017230637872548050513149380284237945835918775705402206362318726395792060727164690207628775840715375203606964965
Підпис підтверджено.


#check_task
pubkey_serv = (int('B147F36F350E32B82562545E92A9D82F5C86A1F02F908C93CD928BAD171647B9', 16), int('10001', 16))
message = 'KittyKitty'
ciphertext_serv = (int('8D0207343313143C514681B7BCA9561458FCAE5CAB04B7777555FF8353C2D95E', 16))
def Encrypt(message, pubkey):
# Конвертуємо повідомлення у числове представлення
m_int = int.from_bytes(message.encode('utf-8'), 'big')
# Шифруємо повідомлення використовуючи відкритий ключ
c_int = pow(m_int, pubkey[1], pubkey[0])
return c_int
ciphertext_our = Encrypt(message, pubkey_serv)
print(f"ciphertext encrypted by server: {ciphertext_serv}\nciphertext encrypted by us: {ciphertext_our}")
#Приклад виведення розшифрованого повідомлення у шістнадцятковому форматі (для сервера для розшифровки)
#print(hex(152767832730931252960509088844401314747835178068421855344631264036079154485))
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Підпис підтверджено.
k = 3019434326298823064033099353236767090223276806879897779851646139320054754877369151657665797196912789896388411330210592455741740724596372376021158806814793819407094029369569703737922556119437947669030313069408008587346485193244326174931941973981813342356422017748199200858535309360120521626831719103386821389592391063296582950671914383508811237648527069200863972983072448324750850157402524129184244995781388464458677205477791962719946606045622882346074488706096436742806370619649611992940941330745835798811371401
Підпис = 1500091579601589134875946297743715584882091113605303948989288155283597246309289774583949039336633928477452224546152877679688809473459064756268971128815970052160954242595612328255232832189875024430767655899067268992082183284478032867732263121574545705395111557867146713413661686841752473011068110764728440352622793474163517948829061376510413210863051250444737834823438050796962435518949277473735164580746221697534091126046794393503265779334053434023004740099900342600686625898609607064141262301489905278656586448
Зашифрований підпис = 4302087048473146782999364739278466044860352149464877186571721958796235807120761464029411647573840410026830665744852477071388042935028876024681534268509084874751760154534163800821984357989349692134896631055751007145764146935760723612678145573101106494233307469133149682579512080563515577182360825524735120044738602379688081799862924152898554818967442639455006168606224580607533266356055332138619418255977349225096139712578349142714190426201023308582641820444560086456178262694077198844810836070784187668342465 3
Зашифрований ключ = 2560564363107481228658290696766248714028257595778654631479599563340275094648474605160525237140261433799205145739927875878320104110259662354607521727875905276635052455883351930353424337292379414514745392343785728151530286773598774421082820191463809279357617995032984468813908360084864180700355149393233573005139879276858065939547746773769057596111727677900433013384974587548425012198491510179675294400770872911802446885254671694265388678183312804475779560937174546961488755625151017430207127673249503715758141863 4
Розшифрований ключ = 1139324650957940498272766928934565075254024763008217272504871567994716872145550043736777851309966443693811032374668856194232937133363883805934348189307479116055066189407371760373016964306994607484063565840504043116903014359000751099039108701226333416655031594444680678021031322102377588689162175532980084701654928277484398957880915863223356860049307627718731012422993937879888364539836787937596965995807621835170425164774290685519855149836452122507067827160519224331271892669779774833870631442517339440017368 6
Розшифрований підпис = 1500091579601589134875946297743715584882091113605303948989288155283597246309289774583949039336633928477452224546152877679688809473459064756268971128815970052160954242595612328255232832189887502443876765589906726899208218328447803286773226312157454570539511557867146713413661686841752473011068110764728440352622793474163517948829061376510413210863051250444737834823438050796962435518949277473731645807462216797534091126046794393503265779334053434023004740099900342600686625898609607064141262301489905278656586448
Підпис не підтверджено.
PS C:\Users\Support\Documents\study\fb-14_khashcha_lab4> python -u "c:\Users\Support\Documents\study\fb-14_khashcha_lab4\1.py"
ciphertext encrypted by server: 63779695063890547294201203523209522894327922914962929363536071045805448026462
ciphertext encrypted by us: 63779695063890547294201203523209522894327922914962929363536071045805448026462
PS C:\Users\Support\Documents\study\fb-14_khashcha_lab4>
Ln 193, Col 2
Spaces: 4
UTF-8
CRLF
Python
3.12.1 64-bit (Microsoft St


Ciphertext
8D0207343313143C514681B7BCA9561458FCAE5CAB04B7777555FF8353C2D9
Text
Decrypt
Message
KittyKitty


Message
KittyKitty
Text
Sign
Signature
1ACFD7926878FC613A99166C69190373A008EE772BD6218EB3EC2C24B7642661




```python
signed = (message, int("1ACFD7926878FC613A99166C69190373A008EE772BD6218EB3EC2C24B7642661", 16))

def Verify(signed, pubkey):
    # Розбиваємо підпис на повідомлення та цифровий підпис
    message, signature = signed
    # Конвертуємо повідомлення у числове представлення
    m_int = int.from_bytes(message.encode('utf-8'), 'big')
    # Верифікуємо цифровий підпис
    verified_signature = pow(signature, pubkey[1], pubkey[0])
    # Якщо числове представлення повідомлення відповідає верифікованому підпису, підпис дійсний
    return m_int == verified_signature

verified = Verify(signed, pubkey_serv)

print(verified)
```

```
ROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

S C:\Users\Support\Documents\study\fb-14_khashcha_lab4> python -u "c:\Users\Support\Documents\study\fb-14_khashcha_lab4\1.py"
iphertext encrypted by server: 63779695063890547294201203523209522894327922914962929363536071045805448026462
iphertext encrypted by us: 63779695063890547294201203523209522894327922914962929363536071045805448026462
rue
S C:\Users\Support\Documents\study\fb-14_khashcha_lab4>
```

Під час лабораторної роботи ознайомився з тестами перевірки чисел на простоту і методами генерації ключів для асиметричної криптосистеми типу RSA. Крім того, ми на практиці ознайомились з системою захисту інформації на основі криптосхеми RSA, організували з використанням цієї системи засекречений зв'язок й електронний підпис, вивчивши протокол розсилання ключів.