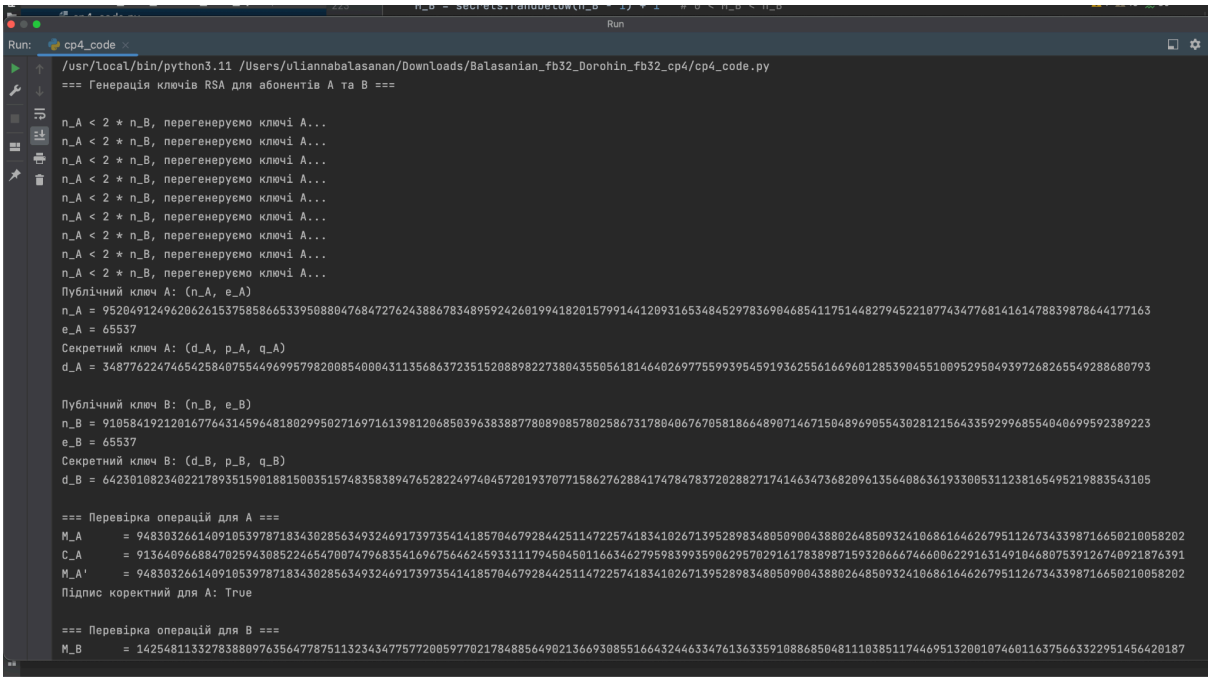
Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут

Криптографія

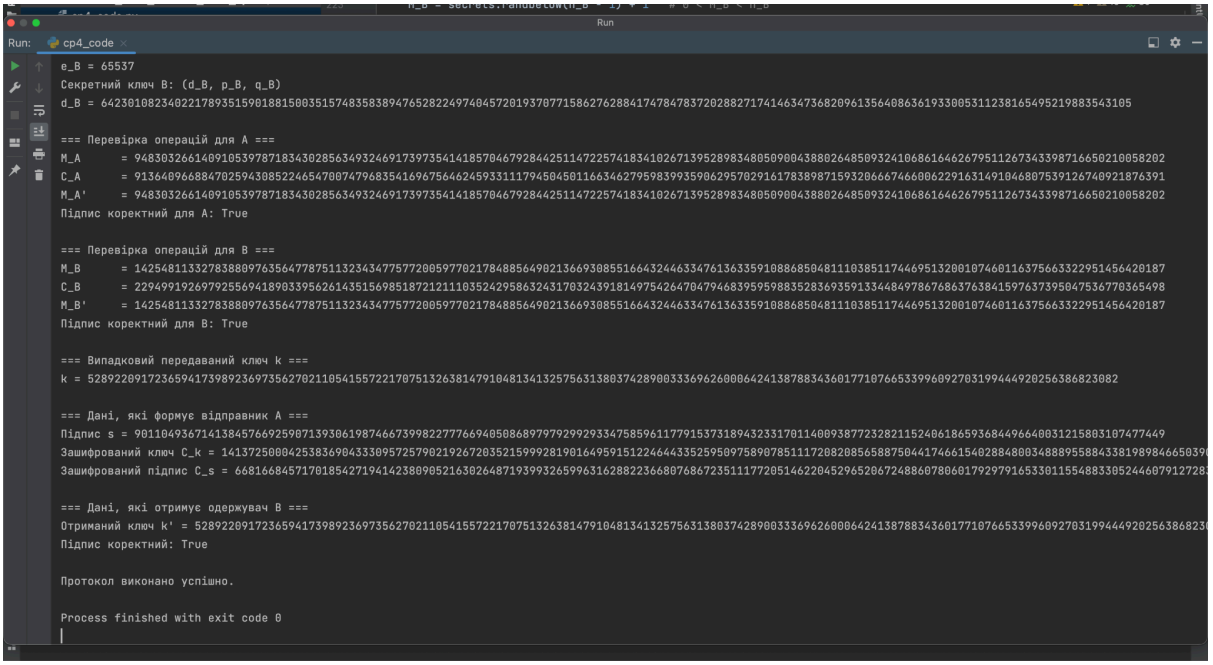
Комп’ютерний практикум №4

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

Виконали:
Студенти 3 курсу
ФБ-32 Баласанян Юліана та
ФБ-32 Дорогін Артем

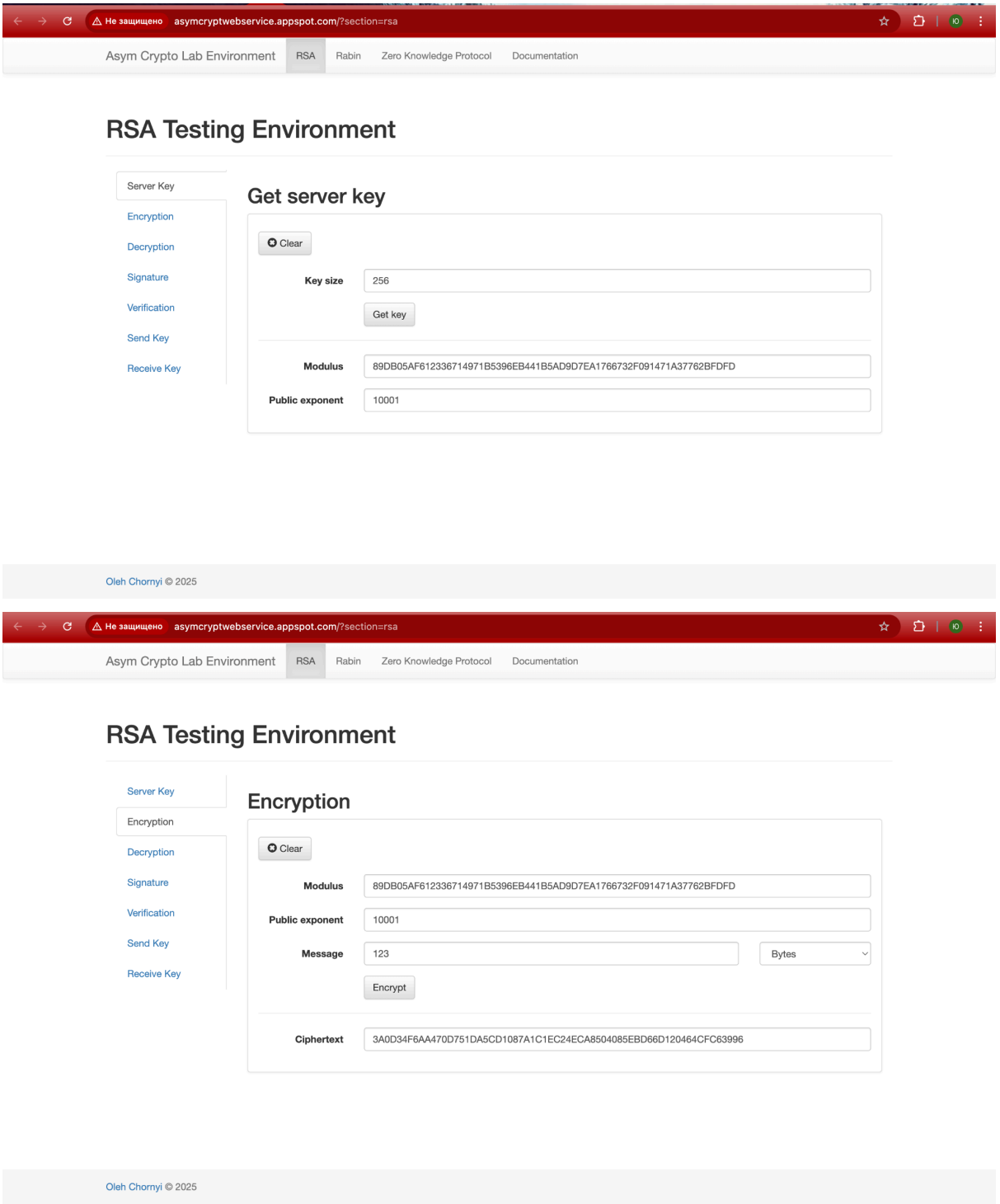


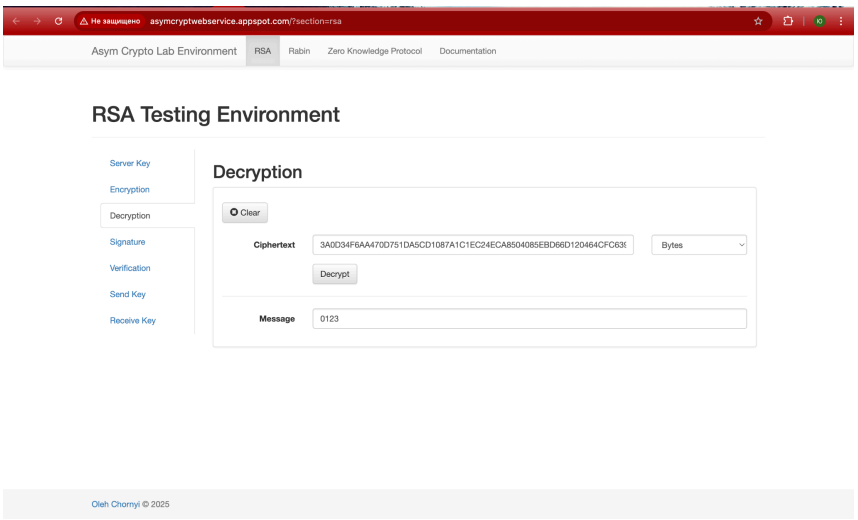
```
Run
Run: cp4_code
/usr/local/bin/python3.11 /Users/uliannabalasanan/Downloads/Balasanian_fb32_Dorohin_fb32_cp4/cp4_code.py
=== Генерація ключів RSA для абонентів A та B ===

n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
n_A < 2 * n_B, перегенеруємо ключі A...
Публічний ключ A: (n_A, e_A)
n_A = 952049124962062615375858665339508804768472762438867834895924260199418201579914412093165348452978369046854117514482794522107743477681416147883987864417716 3
e_A = 65537
Секретний ключ A: (d_A, p_A, q_A)
d_A = 34877622474654258407554496995798200854000431135686372351520889822738043550561814640269775599395459193625561669601285390455100952950493972682655492886807 93

Публічний ключ B: (n_B, e_B)
n_B = 91058419212016776431459648180299502716971613981206850396383887780890857802586731780406767058186648907146715048969055430281215643359299685540406995923892 23
e_B = 65537
Секретний ключ B: (d_B, p_B, q_B)
d_B = 64230108234022178935159018815003515748358389476528224974045720193707715862762884174784783720288271741463473682096135640863619330053112381654952198835431 05

=== Перевірка операцій для A ===
M_A      = 948303266140910539787183430285634932469173973541418570467928442511472257418341026713952898348050900438802648509324106861646267951126734339871665021005820 2
C_A      = 913640966884702594308522465470074796835416967564624593311179450450116634627959839935906295702916178389871593206667466006229163149104680753912674092187639 1
M_A'     = 948303266140910539787183430285634932469173973541418570467928442511472257418341026713952898348050900438802648509324106861646267951126734339871665021005820 2
Підпис коректний для A: True

=== Перевірка операцій для B ===
M_B      = 142548113327838809763564778751132343477577200597702178488564902136693085516643244633476136335910886850481110385117446951320010746011637566332295145642018 7
```

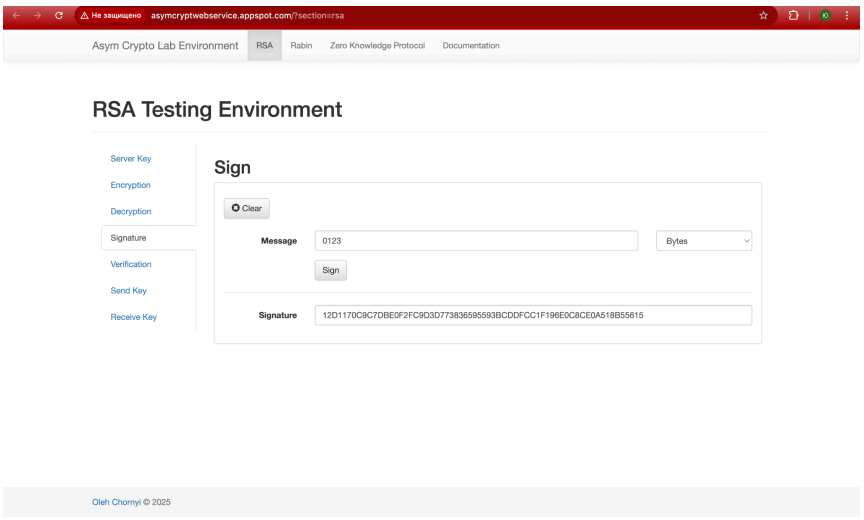


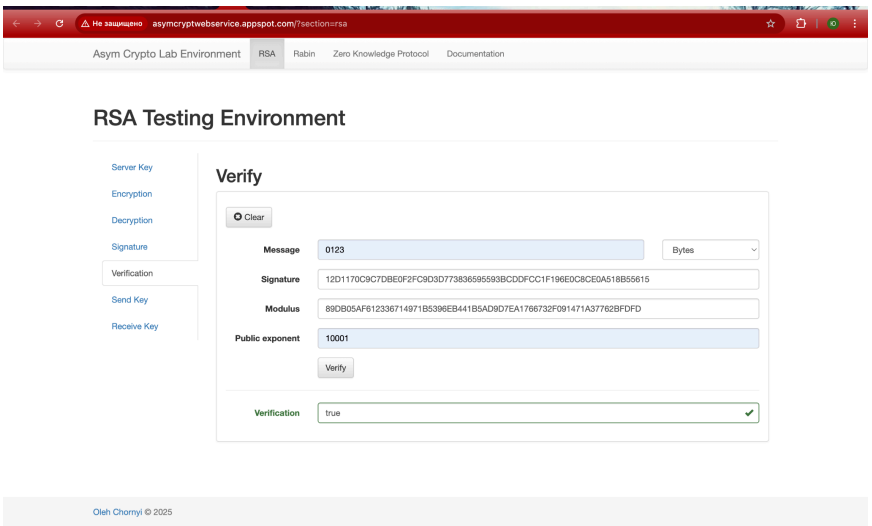
```
Run
Run: cp4_code
e_B = 65537
Секретний ключ B: (d_B, p_B, q_B)
d_B = 64230108234022178935159018815003515748358389476528224974045720193707715862762884174784783720288271741463473682096135640863619330053112381654952198835431 05

=== Перевірка операцій для A ===
M_A      = 948303266140910539787183430285634932469173973541418570467928442511472257418341026713952898348050900438802648509324106861646267951126734339871665021005820 2
C_A      = 913640966884702594308522465470074796835416967564624593311179450450116634627959839935906295702916178389871593206667466006229163149104680753912674092187639 1
M_A'     = 948303266140910539787183430285634932469173973541418570467928442511472257418341026713952898348050900438802648509324106861646267951126734339871665021005820 2
Підпис коректний для A: True

=== Перевірка операцій для B ===
M_B      = 142548113327838809763564778751132343477577200597702178488564902136693085516643244633476136335910886850481110385117446951320010746011637566332295145642018 7
C_B      = 229499192697925569418903395626143515698518721211103524295863243170324391814975426470479468395959883528369359133448497867686376384159763739504753677036549 8
M_B'     = 142548113327838809763564778751132343477577200597702178488564902136693085516643244633476136335910886850481110385117446951320010746011637566332295145642018 7
Підпис коректний для B: True

=== Випадковий передаваний ключ k ===
k = 528922091723659417398923697356270211054155722170751326381479104813413257563138037428900333696260006424138788343601771076653399609270319944492025638682308 2

=== Дані, які формує відправник A ===
Підпис s = 90110493671413845766925907139306198746673998227776694058686979792992933475859611779153731894323317011400938772328211524061865936844966400312158031074774 9
Зашифрований ключ C_k = 141372500042538369043330957257902192672035215999281901649591512246443352595097589078511172082085656887504417466154028848003488895588433819898466503 0
Зашифрований підпис C_s = 668166845717018542719414238090521630264871939932659963162882236680768672351117720514622045296520672488607806017929791653301155488330524460791272 3

=== Дані, які отримує одержувач B ===
Отриманий ключ k' = 528922091723659417398923697356270211054155722170751326381479104813413257563138037428900333696260006424138788343601771076653399609270319944492025638682 3
Підпис коректний: True

Протокол виконано успішно.

Process finished with exit code 0
```

перевірка коректності власної реалізації RSA шляхом взаємодії з тестовим сервером:









Висновок: Для перевірки коректності реалізації RSA ми використали тестовий сервер з методичних рекомендацій. Отримали відкритий ключ сервера, зашифрували повідомлення та перевірили, що сервер коректно його розшифровує. Також ми створили цифровий підпис і перевірили його на сервері. Отримані результати збігаються з

результатами нашої програми, що підтверджує правильність реалізації алгоритмів RSA. Ще у ході роботи було реалізовано повну схему RSA: генерацію випадкових простих чисел, побудову відкритих і закритих ключів, шифрування та розшифрування, формування і перевірку цифрового підпису, а також протокол конфіденційного передавання сесійного ключа