Міністерство освіти і науки України

Національний технічний університет України

"Київський політехнічний інститут імені Ігоря Сікорського"

Фізико-технічний інститут

«Криптографія»

Лабораторна робота №4.

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

Виконали:

Варіант - **5**

студенти гр. ФБ-92

Кудряшов М.О. та Курганський Л. С.

Мета 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.

Загальний код програми знаходиться в файлі "main.py"

Хід роботи:

1. При запуску програма створює двох абонентів "A" і "B", між якими передаються повідомлення;
2. При створенні абонентів, генеруються псевдопрості числа, які будуть використані для шифрування;
3. Абонент "A" генерує повідомлення для "B" для цього йому потрібно:
   - **3.1.** Згенерувати відкритий ключ, котрий менший чим у "B" $n < n1$;
   - **3.2.** Закодувати і зашифрувати повідомлення;
   - **3.3.** Згенерувати підпис.
4. Абонент "B" отримує пакет даних з повідомленням і підписом, та відкритий ключ "A";
5. Розшифрування повідомлення абонентом "B":
   - **5.1.** Розшифровує повідомлення своїм секретним(d) і відкритим ключем (n);
   - **5.2.** Розшифровує підпис своїм секретним(d) і відкритим ключем (n);
   - **5.3.** Перевірка підпису з повідомленням;
   - **5.4.** Вивід повідомлення або помилки;
6. Дії повторюються поки листування не закінчиться.

Приклад роботи:

```python
A = Person()
B = Person()

text = "lol kek cheburek."

packet = A.create_message(text, B.open_key)
print(f"msg={packet[0]}\nsign={packet[1]}\n")
B.read_message(packet, A.open_key)

text = "Ok, Da."

packet = B.create_message(text, A.open_key)
print(f"msg={packet[0]}\nsign={packet[1]}\n")
A.read_message(packet, B.open_key)
```

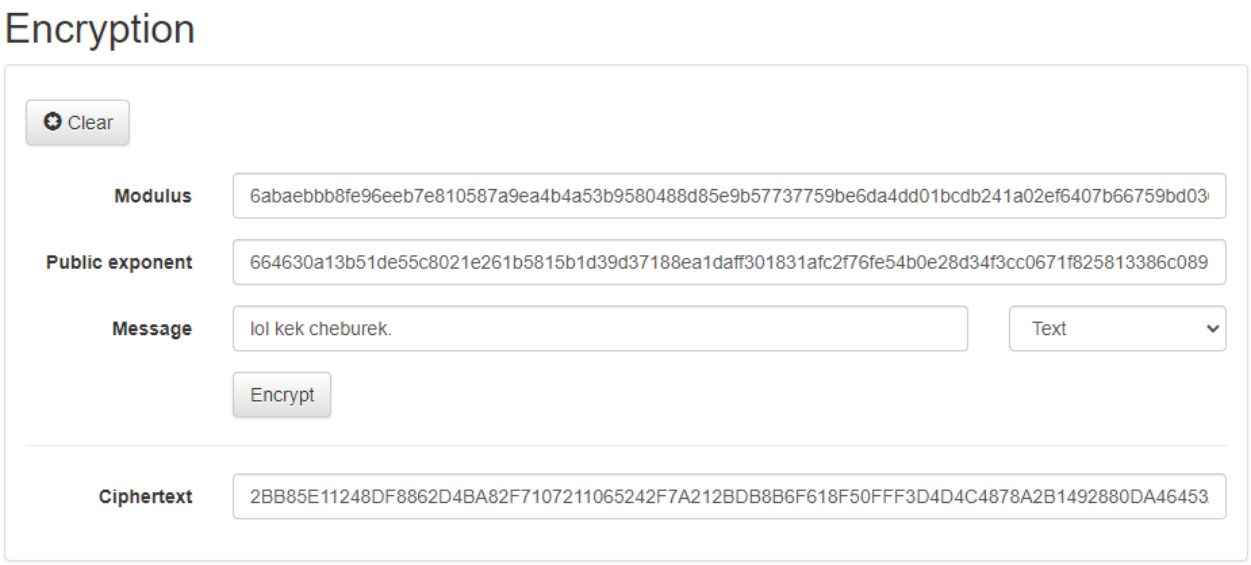
Результат:

```
PS C:\Users\lkurg\Desktop\ФІЗТЕХ 3.1\crypta\lab_4> & C:/Users/lkurg/AppData/Local/Programs/Python/Python39/python.exe "C:\Users\lkurg\Desktop\Ф?ЗТЕХ 3.1\crypta\lab_4\main.py"
msg=4870690465432927738642461802068979465980722408763435979713096946447014971122096307391057886163506718079509494778637648723509777770599754457664983142825264
sign=391362288509395444310012111460003553265236041942569345442006929094596920669064029539982004964188583970082077977018184217619835967015768022802699384683965 9

message: "lol kek cheburek."
msg=526564882729483027879876318720597767222263979662912603007216216938816730800742397324635765125333827161004488759897908040562989554603721707789306148719885 7
sign=361656487543987166662683511674304291680949782575010118518238375837349046426927578494297950245349061778558281660744807802861606831337667695459555971457955 1

message: "Ok, Da."
PS C:\Users\lkurg\Desktop\ФІЗТЕХ 3.1\crypta\lab_4>
```

Перевірка сайтом:



```
text => lol kek cheburek.
Modulus => 6abaebbb8fe96eeb7e810587a9ea4b4a53b9580488d85e9b57737759be6da4dd01bcdb241a02ef6407b66759bd036d2de1732b168a45d4b1ebd5f62fb81c82bb
Public exponent => 664630a13b51de55c8021e261b5815b1d39d37188ea1daff301831afc2f76fe54b0e28d34f3cc0671f825813386c089b00403868fc56df03423843afb77922ad

Ciphertext=2bb85e11248df8862d4ba82f7107211065242f7a212bdb8b6f618f50fff3d4d4c4878a2b1492880da46453ae67feb06a2cbafa214363da942547dcbe83b640fd
PS C:\Users\lkurg\Desktop\ФІЗТЕХ 3.1\crypta\lab_4>
```