

## Криптографія

### Комп'ютерний практикум №4

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

Варіант 10

Виконав: студент групи ФБ-91  
Кузавка Артем

Київ – 2021

## **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

## Завдання

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  $pq \leq p_1 q_1$ ;  $p$  і  $q$  – прості числа для побудови ключів абонента А,  $p_1$  і  $q_1$  – абонента В.
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 А і В – 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 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  $M$  і знайти криптограму для абонентів А і В, перевірити правильність розшифрування. Скласти для А і В 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

Було написано програму “Miller-Rabin.py” що генерує велике число (256) біт та перевіряє його на простоту за допомогою теста Міллера-Рабіна.

Демонстрація роботи програми:

```
nyog-sothep@PC:~/Desktop/Crypto/lab_4$ ./Miller-Rabin.py  
Newly generated prime number is: 65103311422218740613215361075078750889974688739419206771128497690366045861319
```

Рисунок 1: Згенероване просте число, перевірене алгоритмом Міллера-Рабіна

Кандидати, що не пройшли перевірку:

|   |
|---|
| 99614217205754232967211858024284175428116220829249854766533070928249867716672 |
| 55607185479069846360047376172377805719040504422768053522363283122864541783953 |
| 71146815524895807960335074536940630885095082182436030208700148166793745516167 |
| 59772929079763699758125941783597390273127576639668873916512147170724749559641 |

Кандидати, що пройшли перевірку:

|   |
|---|
| 5406180505134767480590370748207604657532748231574128423436936460711860925357513 |
| 2858362201169014458608392208377457198124059682172475213728075095338001691394017 |
| 11607884477518636648231113472655406550460960133193341352495404868807879111319   |
| 59772929079763699758125941783597390273127576639668873916512147170724749559621   |

Було реалізовано алгоритм шифрування, розшифрування, підпису та перевірки підпису в програмі “RSA.py”.

Відкритий текст наведено нижче.

```

(                                     ,&&&.
)                                     ,.,&&
( (                                  \=_/_
)                                     ,'-'.
( ( ( ,,- _._|/_/_
) /\ -((-----((_|___/_|
( //_|(`' (( `'-_|
_ -.,;/_ \ \ -._ _ \ \ \ -._/_ .
(_; -//| \ \ -' . \ <_, \ \ `--'|
( `._ _ _ ,') <_, -' _ , '
`'(_ )_)( _ )_)'

```

У числовому форматі:

32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 40, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32  
 , 32, 32, 32, 44, 38, 38, 38, 46, 10, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 41, 32, 32, 3  
 2, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 46, 44, 46, 38, 38, 10, 32, 32, 32, 32, 32,  
 32, 32, 32, 32, 32, 32, 40, 32, 32, 40, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 92  
 , 61, 95, 95, 47, 10, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 41, 32, 32, 32, 3  
 2, 32, 32, 32, 32, 32, 32, 32, 32, 32, 44, 39, 45, 39, 46, 10, 32, 32, 32, 32, 32, 32, 32, 32, 32,  
 40, 32, 32, 32, 32, 40, 32, 32, 44, 44, 32, 32, 32, 32, 32, 32, 95, 46, 95, 95, 124, 47, 32, 47, 1  
 24, 10, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 41, 32, 47, 92, 32, 45, 40, 40, 45, 45, 45, 45,  
 45, 40, 40, 95, 124, 95, 95, 95, 47, 32, 124, 10, 32, 32, 32, 32, 32, 32, 32, 32, 40, 32, 32, 47  
 , 47, 32, 124, 32, 40, 96, 39, 32, 32, 32, 32, 32, 32, 40, 40, 32, 32, 96, 39, 45, 45, 124, 10, 32  
 , 32, 32, 32, 32, 32, 95, 32, 45, 46, 59, 95, 47, 32, 92, 45, 45, 46, 95, 32, 32, 32, 32, 32, 9  
 2, 32, 92, 45, 46, 95, 47, 46, 10, 32, 32, 32, 32, 32, 40, 95, 59, 45, 47, 47, 32, 124, 32, 92, 32  
 , 92, 45, 39, 46, 92, 32, 32, 32, 32, 60, 95, 44, 92, 95, 92, 96, 45, 45, 39, 124, 10, 32, 32, 32,  
 32, 32, 40, 32, 96, 46, 95, 95, 32, 95, 32, 32, 95, 95, 95, 44, 39, 41, 32, 32, 32, 32, 32, 32, 60  
 , 95, 44, 45, 39, 95, 95, 44, 39, 10, 32, 32, 32, 32, 32, 32, 96, 39, 40, 95, 32, 41, 95, 41, 40, 9  
 5, 41, 95, 41, 39

Ключ, яким було зашифровано повідомлення та ключ для його розшифрування відповідно наведені нижче.

Файл “A.enc”:

17,15452822008574028448658113950346810956974576021332104601930514562982110  
22273019849086006444997581928191807287931784074289302398127924363640670447  
9781074199721

Файл "A.dec":

```
10907874358993431846111609847303631263746759544469720895480363220928548392
51542839504754692672509182340361150182775417841301898918278103863186947994
2440551665,154528220085740284486581139503468109569745760213321046019305145
62982110222730198490860064449975819281918072879317840742893023981279243636
406704479781074199721
```

Шифротекст наведено в [додатку А](#).

Деякі дані разом із сигнатурою:

```
Artem Kuzavka --
37166241462974278584796992491942989434912300697696693151628405717819484520
158879812718529426796359012673563079141576651901298573
```

Для їх підпису було використано ключ A\_епс, що виступав у ролі приватного.

Отриманий підпис: наведено в [додатку Б](#).

У ході роботи програми було згенеровано ключі, зашифровано та розшифровано текст, підписано дані й перевірено підпис.

Вивід програми:

```
nyog-sothep@PC:~/Desktop/Crypto/lab_4$ ./RSA.py
Decrypted text is:
      ( ,&&&.
      ) ..&&
      ( ( \=_/_/
      ) '._'
      ( ( ,,, _/_/_/
      ) /\ -((-----(_/_/_/
      ( // | (' (( '_--|
      _-;/_ \-_- _ \ \-_/
      (-; -// | \ \-' \ <_,\ \-''|
      ( :_-_-_-_) <_-''_-
      ''(-_-)(-_-)'
```

The signature was successfully verified: True

Рисунок 2: Вивід програми "RSA.py"

Перевірка розшифрування зашифрованого тексту (літери “А”) онлайн на ресурсі [www.decode.fr/rsa-cipher](http://www.decode.fr/rsa-cipher)



← → ↻ 🏠 🔒 https://www.dcode.fr/rsa-cipher



Search for a tool

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

★ BROWSE THE FULL DCODE TOOLS' LIST

**Results**

🚩 ✓ Déryption using C,D,N

A

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=

6599743590836592050933837890625

★ PUBLIC KEY E (USUALLY E=65537) E=

17

★ PUBLIC KEY VALUE (INTEGER) N=

15452822008574028448658113950346810956974576021332

★ PRIVATE KEY VALUE (INTEGER) D=

10907874358993431846111609847303631263746759544469

★ FACTOR 1 (PRIME NUMBER) P=

54061805051347674805903707482076046575327482315741

★ FACTOR 2 (PRIME NUMBER) Q=

28583622011690144586083922083774571981240596821724

★ 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**

Рисунок 3: Успішне розшифрування локально зашифрованого тексту на сторонньому ресурсі

## Тестування на Asym Crypto Lab Environment

The screenshot shows the 'Get server key' interface of the Asym Crypto Lab Environment. The browser address bar shows the URL `https://asym-crypt-study.herokuapp.com/?section=rsa`. The navigation bar includes 'Asym Crypto Lab Environment', 'RSA' (selected), 'Rabin', 'Zero Knowledge Protocol', and 'Documentation'. The left sidebar contains links: 'Server Key' (selected), 'Encryption', 'Decryption', 'Signature', 'Verification', 'Send Key', and 'Receive Key'. The main content area is titled 'Get server key' and contains a 'Clear' button, a 'Key size' input field with the value '512', a 'Get key' button, a 'Modulus' input field with a long hexadecimal string, and a 'Public exponent' input field with the value '10001'.

Asym Crypto Lab Environment RSA Rabin Zero Knowledge Protocol Documentation

### RSA Testing Environment

Server Key Encryption Decryption Signature Verification Send Key Receive Key

#### Get server key

Clear

Key size 512

Get key

Modulus D978B78BFCE79B1C3C626E3F72833C4EFBCE0F649608A73D7A0B9A5BF774B8C4A1C:

Public exponent 10001

Рисунок 4: Ger server key

The screenshot shows the 'Receive key' interface of the Asym Crypto Lab Environment. The browser address bar shows the URL `https://asym-crypt-study.herokuapp.com/?section=rsa#encryption_tab`. The navigation bar is the same as in the previous screenshot. The left sidebar contains links: 'Server Key', 'Encryption', 'Decryption', 'Signature', 'Verification', 'Send Key', and 'Receive Key' (selected). The main content area is titled 'Receive key' and contains a 'Clear' button, four input fields for 'Key', 'Signature', 'Modulus', and 'Public exponent' (all containing long hexadecimal strings), a 'Receive' button, another 'Key' input field, and a 'Verification' input field with the value 'true' and a green checkmark icon.

Asym Crypto Lab Environment RSA Rabin Zero Knowledge Protocol Documentation

### RSA Testing Environment

Server Key Encryption Decryption Signature Verification Send Key Receive Key

#### Receive key

Clear

Key 36186bc2894c33fe803f836b995fe0a5416b24b03e2f889899c61f7941792297f9b76d04758fe

Signature 4b93ee5de475b12d44e85de27f5cf8123e48d82bce63a94d4a915515839e1602c87a73fa568

Modulus 250769e9779a3e45f90424d6cb17704e9fa27624575433ff1129b0dc1b4584bef7c3e49ab14a

Public exponent 49e42ba5aafa872d45e861fbfd5a5e19084b8328498ec169eca3e28b6a713a5a19f430d6ab0

Receive

Key 21463EE95A84B9581E454412958B0CE252AC4DF0D11FCF6B72EA693B71C6183342991

Verification true

Рисунок 5: Receive key



## **Висновки**

В ході виконання практичного завдання я зіткнувся з труднощами перевірки великого числа на простоту. Проблема частково була вирішена використанням тесту Рабіна-Міллера, остаточно вдалося вирішити дану проблему використанням даного тесту в після перевірки даного числа на подільність меншими числами з трьома й манше цифрами.

Я навчився реалізовувати й працювати з алгоритмом асиметричного шифрування RSA та дізнався про складнощі перевірки числа на простоту.

## Додаток А: шифротекст

[illegible]

0785295216663082172416, 10000000000000000000, 38685626227668133590597632, 38685  
626227668133590597632, 38685626227668133590597632, 38685626227668133590597632,  
, 38685626227668133590597632, 38685626227668133590597632, 38685626227668133590  
597632, 38685626227668133590597632, 38685626227668133590597632, 17179869184000  
00000000000000, 38685626227668133590597632, 38685626227668133590597632, 386856  
26227668133590597632, 38685626227668133590597632, 1717986918400000000000000000  
0, 38685626227668133590597632, 38685626227668133590597632, 8683513829059386822  
166052864, 8683513829059386822166052864, 38685626227668133590597632, 386856262  
27668133590597632, 38685626227668133590597632, 38685626227668133590597632, 386  
85626227668133590597632, 38685626227668133590597632, 418120335219177412867660  
5224609375, 18487710785295216663082172416, 4181203352191774128676605224609375  
, 4181203352191774128676605224609375, 387408056264039998936360748396314624, 26  
647936506962193439322192687, 38685626227668133590597632, 26647936506962193439  
322192687, 387408056264039998936360748396314624, 10000000000000000000, 386856262  
27668133590597632, 38685626227668133590597632, 38685626227668133590597632, 386  
85626227668133590597632, 38685626227668133590597632, 386856262276681335905976  
32, 38685626227668133590597632, 38685626227668133590597632, 386856262276681335  
90597632, 38685626227668133590597632, 2614120267500775228203738281, 3868562622  
7668133590597632, 26647936506962193439322192687, 2423221228050214638463506502  
909952, 38685626227668133590597632, 12723679885609870147705078125, 17179869184  
000000000000000000, 171798691840000000000000000000, 1272367988560987014770507812  
5, 12723679885609870147705078125, 12723679885609870147705078125, 1272367988560  
9870147705078125, 12723679885609870147705078125, 1272367988560987014770507812  
5, 171798691840000000000000000000, 171798691840000000000000000000, 418120335219177  
4128676605224609375, 387408056264039998936360748396314624, 418120335219177412  
8676605224609375, 4181203352191774128676605224609375, 41812033521917741286766  
05224609375, 26647936506962193439322192687, 38685626227668133590597632, 387408  
056264039998936360748396314624, 10000000000000000000, 3868562622766813359059763  
2, 38685626227668133590597632, 38685626227668133590597632, 3868562622766813359  
0597632, 38685626227668133590597632, 38685626227668133590597632, 3868562622766  
8133590597632, 38685626227668133590597632, 1717986918400000000000000000, 38685  
626227668133590597632, 38685626227668133590597632, 26647936506962193439322192  
687, 26647936506962193439322192687, 38685626227668133590597632, 38740805626403  
9998936360748396314624, 38685626227668133590597632, 17179869184000000000000000  
000, 4995868076798137881795553463894016, 1117116121846700839825703079, 3868562  
6227668133590597632, 38685626227668133590597632, 38685626227668133590597632, 3  
8685626227668133590597632, 38685626227668133590597632, 3868562622766813359059  
7632, 171798691840000000000000000000, 1717986918400000000000000000, 386856262276  
68133590597632, 38685626227668133590597632, 499586807679813788179555346389401  
6, 1117116121846700839825703079, 12723679885609870147705078125, 12723679885609  
870147705078125, 387408056264039998936360748396314624, 10000000000000000000, 386  
85626227668133590597632, 38685626227668133590597632, 386856262276681335905976  
32, 38685626227668133590597632, 38685626227668133590597632, 386856262276681335  
90597632, 4181203352191774128676605224609375, 38685626227668133590597632, 1272  
3679885609870147705078125, 18487710785295216663082172416, 1271991467017507741  
703714391419, 4181203352191774128676605224609375, 266479365069621934393221926  
87, 38685626227668133590597632, 2423221228050214638463506502909952, 1272367988  
5609870147705078125, 12723679885609870147705078125, 1848771078529521666308217  
2416, 4181203352191774128676605224609375, 38685626227668133590597632, 38685626  
227668133590597632, 38685626227668133590597632, 38685626227668133590597632, 38  
685626227668133590597632, 38685626227668133590597632, 24232212280502146384635  
06502909952, 386

128676605224609375, 26647936506962193439322192687, 18487710785295216663082172  
416, 100000000000000000, 38685626227668133590597632, 3868562622766813359059763  
2, 38685626227668133590597632, 38685626227668133590597632, 3868562622766813359  
0597632, 1717986918400000000000000000, 4181203352191774128676605224609375, 127  
1991467017507741703714391419, 12723679885609870147705078125, 2664793650696219  
3439322192687, 26647936506962193439322192687, 38685626227668133590597632, 3874  
08056264039998936360748396314624, 38685626227668133590597632, 242322122805021  
4638463506502909952, 38685626227668133590597632, 2423221228050214638463506502  
909952, 12723679885609870147705078125, 1117116121846700839825703079, 184877107  
85295216663082172416, 2423221228050214638463506502909952, 3868562622766813359  
0597632, 38685626227668133590597632, 38685626227668133590597632, 3868562622766  
8133590597632, 16926659444736000000000000000000, 41812033521917741286766052246  
09375, 8683513829059386822166052864, 2423221228050214638463506502909952, 41812  
03352191774128676605224609375, 2423221228050214638463506502909952, 4995868076  
798137881795553463894016, 12723679885609870147705078125, 12723679885609870147  
705078125, 1117116121846700839825703079, 387408056264039998936360748396314624  
, 10000000000000000000, 38685626227668133590597632, 38685626227668133590597632, 3  
8685626227668133590597632, 38685626227668133590597632, 3868562622766813359059  
7632, 171798691840000000000000000000, 38685626227668133590597632, 49958680767981  
37881795553463894016, 18487710785295216663082172416, 418120335219177412867660  
5224609375, 4181203352191774128676605224609375, 38685626227668133590597632, 41  
81203352191774128676605224609375, 38685626227668133590597632, 386856262276681  
33590597632, 4181203352191774128676605224609375, 4181203352191774128676605224  
609375, 4181203352191774128676605224609375, 8683513829059386822166052864, 1117  
116121846700839825703079, 2614120267500775228203738281, 386856262276681335905  
97632, 38685626227668133590597632, 38685626227668133590597632, 386856262276681  
33590597632, 38685626227668133590597632, 38685626227668133590597632, 169266594  
447360000000000000000000, 4181203352191774128676605224609375, 86835138290593868  
22166052864, 12723679885609870147705078125, 1117116121846700839825703079, 4181  
203352191774128676605224609375, 4181203352191774128676605224609375, 868351382  
9059386822166052864, 1117116121846700839825703079, 10000000000000000000, 3868562  
6227668133590597632, 38685626227668133590597632, 38685626227668133590597632, 3  
8685626227668133590597632, 38685626227668133590597632, 3868562622766813359059  
7632, 4995868076798137881795553463894016, 1117116121846700839825703079, 171798  
691840000000000000000000, 4181203352191774128676605224609375, 38685626227668133  
590597632, 2614120267500775228203738281, 4181203352191774128676605224609375, 2  
614120267500775228203738281, 1717986918400000000000000000, 418120335219177412  
8676605224609375, 2614120267500775228203738281, 41812033521917741286766052246  
09375, 2614120267500775228203738281, 1117116121846700839825703079

## Додаток Б: підписані дані

6599743590836592050933837890625, 92764641967130171567625832766767104, 1246768  
48765984328031674121957933056, 11843044313729355057238118681361701, 432763341  
03547425867991106950436269, 38685626227668133590597632, 751694681821390986442  
56591796875, 144264558065210807467328187211661877, 29384419904780833161828328  
6773235712, 5958260438588051333281183456765537, 16672246556491877472058925271  
2071168, 31588152109649857868144549324788907, 5958260438588051333281183456765  
537, 38685626227668133590597632, 12723679885609870147705078125, 12723679885609  
870147705078125, 38685626227668133590597632, 106829942260164217198710340851, 3  
85625479506907479095458984375, 54116956037952111668959660849, 282288975128239  
507545882230784, 282288975128239507545882230784, 7629394531250000000000000000  
0, 148613013882162475899836956672, 54116956037952111668959660849, 148613013882  
162475899836956672, 282288975128239507545882230784, 7629394531250000000000000000  
0000, 707738052117387173214918768057, 385625479506907479095458984375, 14861301  
3882162475899836956672, 762939453125000000000000000000, 3856254795069074790954  
58984375, 523837348053896201440996622336, 205442259656281392806087233013, 5238  
37348053896201440996622336, 148613013882162475899836956672, 38562547950690747  
9095458984375, 707738052117387173214918768057, 282288975128239507545882230784  
, 707738052117387173214918768057, 707738052117387173214918768057, 762939453125  
000000000000000000, 148613013882162475899836956672, 70773805211738717321491876  
8057, 54116956037952111668959660849, 707738052117387173214918768057, 148613013  
882162475899836956672, 762939453125000000000000000000, 70773805211738717321491  
8768057, 523837348053896201440996622336, 707738052117387173214918768057, 14861  
3013882162475899836956672, 106829942260164217198710340851, 148613013882162475  
899836956672, 707738052117387173214918768057, 54116956037952111668959660849, 7  
629394531250000000000000000000, 106829942260164217198710340851, 381154485839701  
6816554454528, 3811544858397016816554454528, 282288975128239507545882230784  
, 707738052117387173214918768057, 385625479506907479095458984375, 282288975128  
239507545882230784, 707738052117387173214918768057, 2822889751282395075458822  
30784, 282288975128239507545882230784, 707738052117387173214918768057, 1068299  
42260164217198710340851, 54116956037952111668959660849, 205442259656281392806  
087233013, 54116956037952111668959660849, 282288975128239507545882230784, 7629  
39453125000000000000000000, 523837348053896201440996622336, 148613013882162475  
899836956672, 3811544858397016816554454528, 205442259656281392806087233013, 3  
85625479506907479095458984375, 54116956037952111668959660849, 385625479506907  
479095458984375, 523837348053896201440996622336, 5411695603795211166895966084  
9, 707738052117387173214918768057, 148613013882162475899836956672, 52383734805  
3896201440996622336, 148613013882162475899836956672, 205442259656281392806087  
233013, 762939453125000000000000000000, 3811544858397016816554454528, 54116956  
037952111668959660849, 205442259656281392806087233013, 5238373480538962014409  
96622336, 523837348053896201440996622336, 385625479506907479095458984375, 7077  
38052117387173214918768057, 523837348053896201440996622336, 54116956037952111  
668959660849, 762939453125000000000000000000, 385625479506907479095458984375, 5  
4116956037952111668959660849, 523837348053896201440996622336, 205442259656281  
392806087233013, 762939453125000000000000000000, 70773805211738717321491876805  
7, 148613013882162475899836956672, 762939453125000000000000000000, 282288975128  
239507545882230784, 385625479506907479095458984375, 7077380521173871732149187  
68057, 282288975128239507545882230784, 106829942260164217198710340851, 2054422  
59656281392806087233013, 707738052117387173214918768057, 38115448583970168165  
554454528, 54116956037952111668959660849, 762939453125000000000000000000, 28228  
8975128239507545882230784, 385625479506907479095458984375, 106829942260164217

198710340851, 205442259656281392806087233013, 282288975128239507545882230784,  
106829942260164217198710340851, 38115448583970168165554454528, 38562547950690  
7479095458984375, 707738052117387173214918768057, 541169560379521116689596608  
49, 148613013882162475899836956672, 54116956037952111668959660849, 20544225965  
6281392806087233013, 385625479506907479095458984375, 282288975128239507545882  
230784, 282288975128239507545882230784, 205442259656281392806087233013, 541169  
56037952111668959660849, 707738052117387173214918768057, 38115448583970168165  
554454528, 54116956037952111668959660849, 7629394531250000000000000000, 70773  
8052117387173214918768057, 523837348053896201440996622336, 205442259656281392  
806087233013, 385625479506907479095458984375, 106829942260164217198710340851