**ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»  
(СПбГУТ)**

Факультет Инфокоммуникационных сетей и систем

Кафедра Защищенных систем связи

Дисциплина: ОКОК

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 3**

Исследование криптосистемы с открытым ключом РША  
*(тема отчета)*

Направление/специальность подготовки:

11.03.02 Инфокоммуникационные технологии и системы связи

*(код и наименование направления/специальности)*

Студент:

Ковалев И. А., ИКТЗ-73

*(Ф.И.О., № группы) (подпись)*

Проверил:

Проф. Яковлев В.А.

*(Ф.И.О., должность) (подпись)*

**Оглавление**

[Цель работы 3](#_Toc52893707)

[Используемое программное обеспечение 3](#_Toc52893708)

[Задание 3](#_Toc52893709)

[Порядок выполнения лабораторной работы 3](#_Toc52893710)

**Цель работы**

Закрепить знания, полученные на лекциях, по теме “Криптосистема РША”.

**Используемое программное обеспечение**

Для работы используется программа “RSA”.

**Задание**

1.Выполнить упражнения по возведению в степень больших целых чисел по модулю.

2.Произвести генерирование ключей для криптосистемы РША, а также шифрование и дешифрование с ее помощью коротких текстов.

3.Произвести шифрование файлов большого размера безопасной криптосистемой РША и оценить время шифрования и дешифрования при ее простейшей программной реализации.

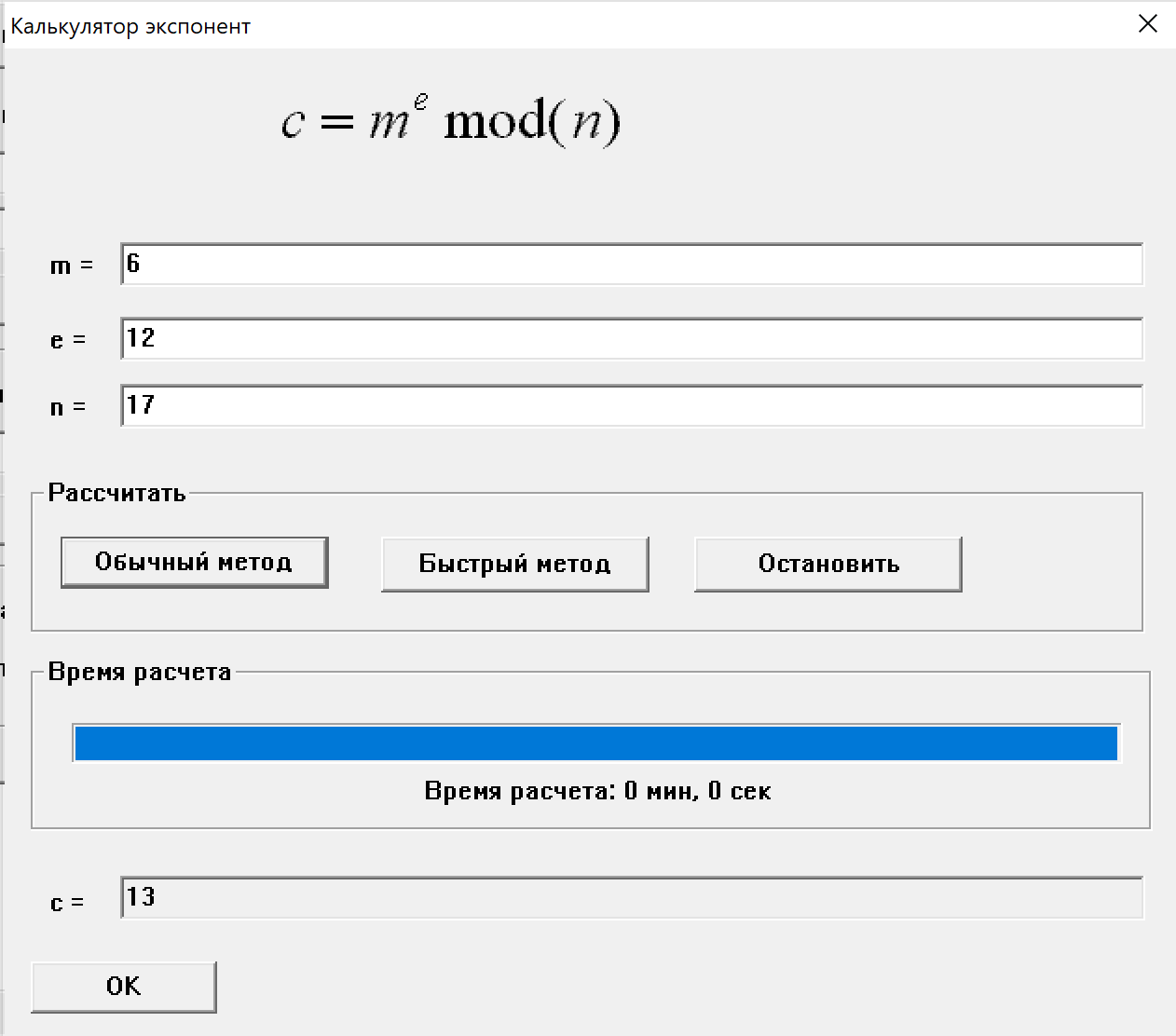
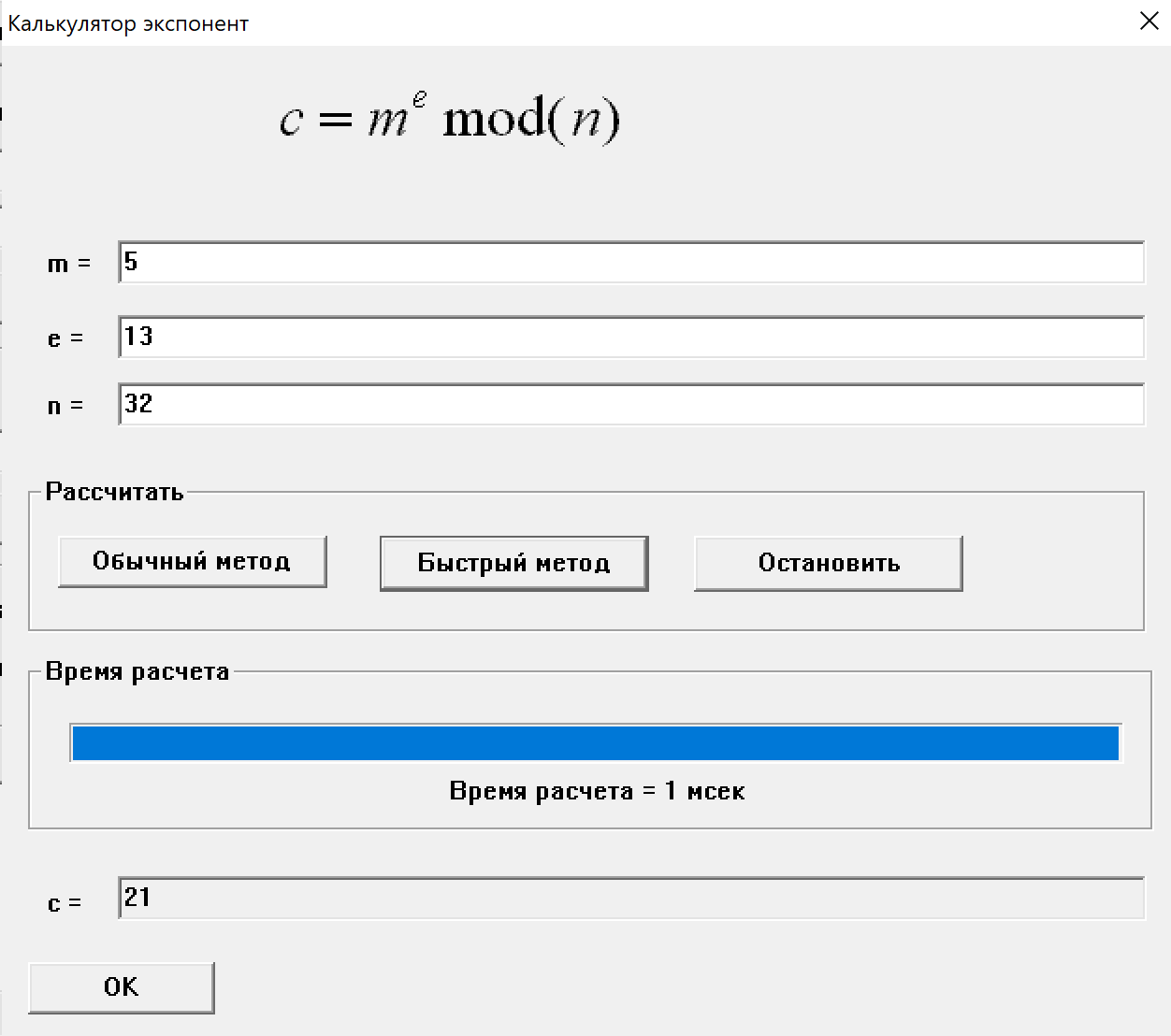
4. Произвести подписание сообщений и проверку подписи.

5. Осуществить обмен зашифрованными и подписанными сообщениями в локальной сети.

**Порядок выполнения лабораторной работы**

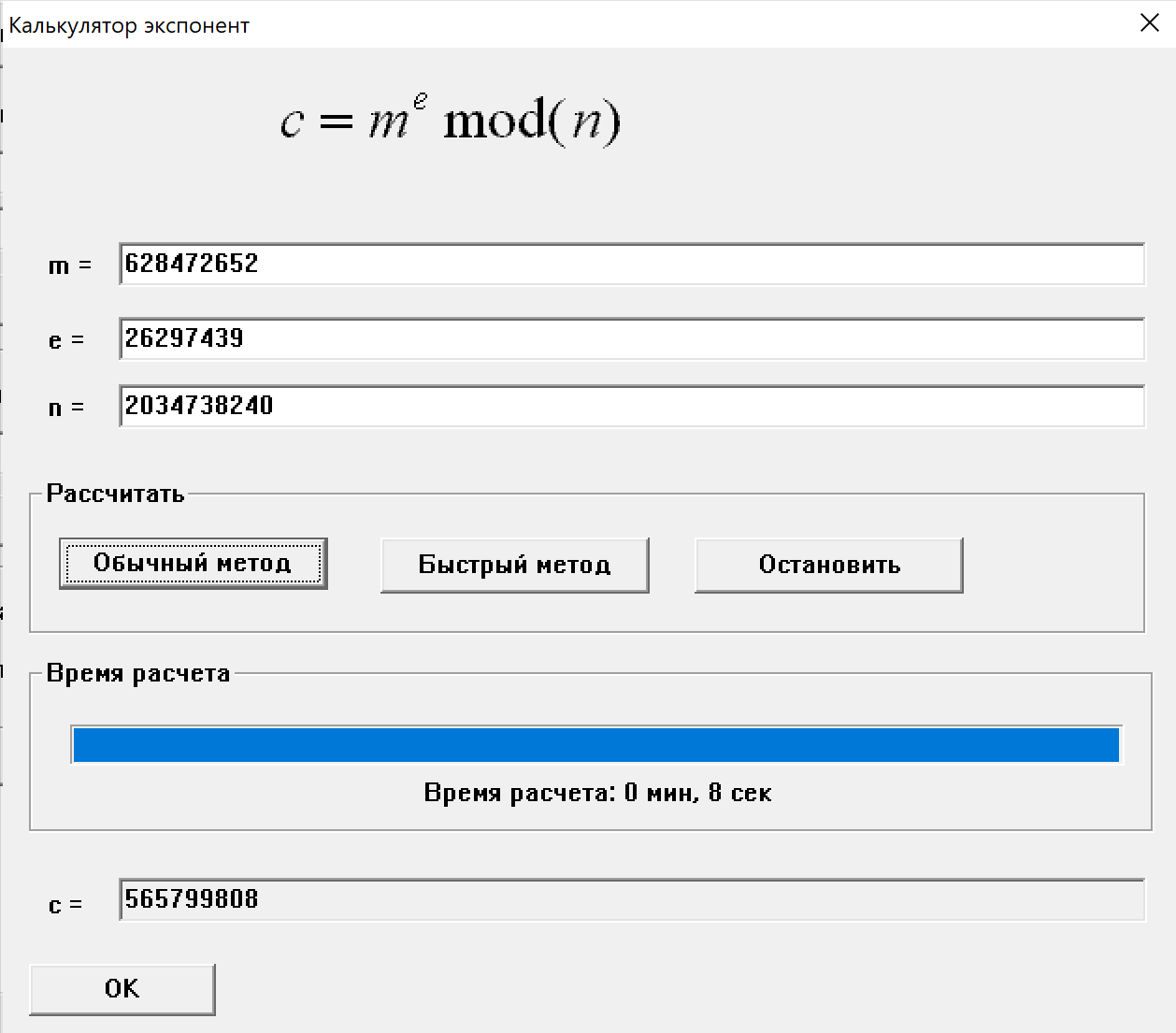
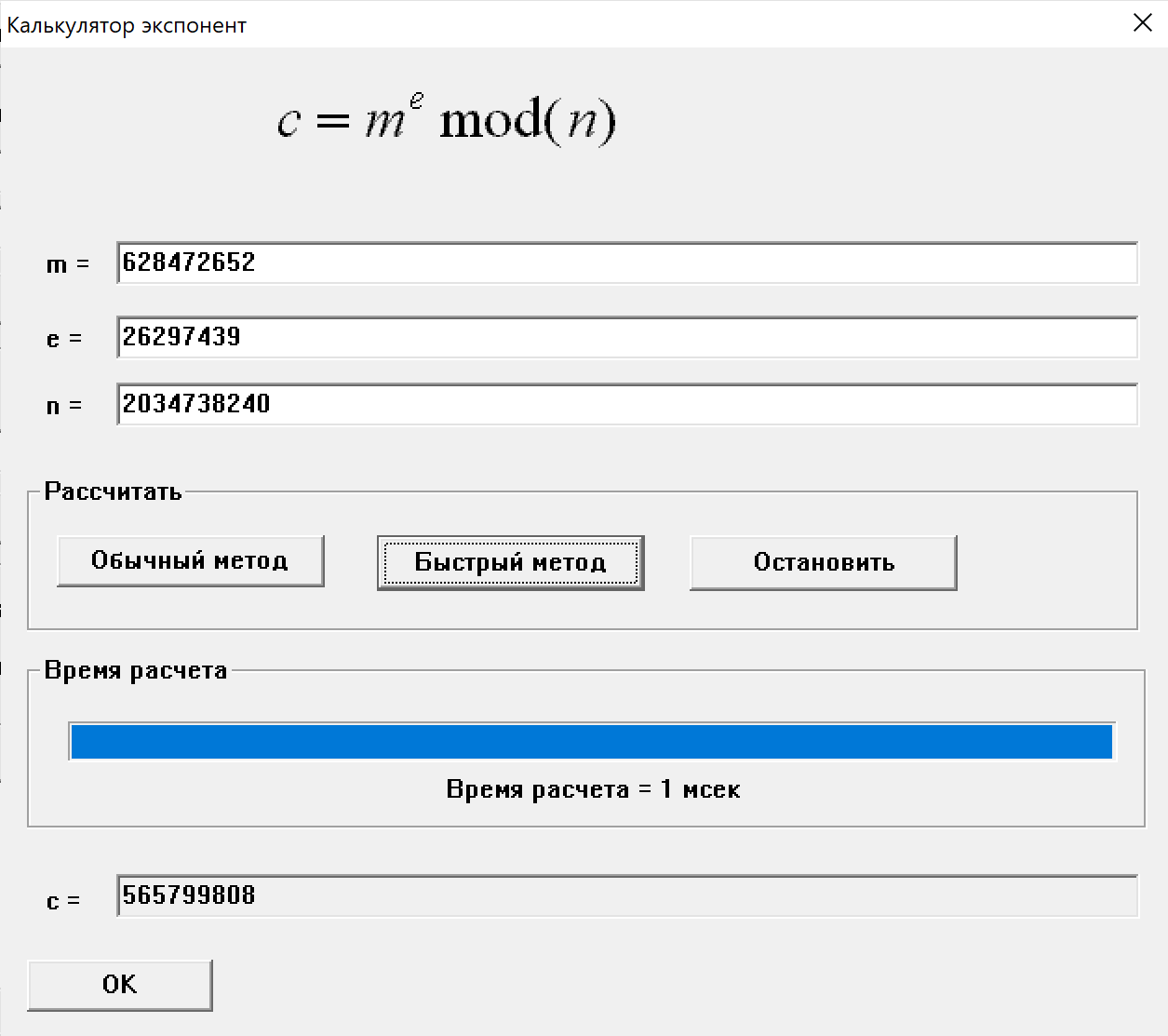
1.Запускаем программу “RSA”.

2.Выбрав в главном меню опцию “Калькулятор экспонент”, рассчитываем степени малых чисел по малому модулю и проверяем их правильность расчетами на бумаге

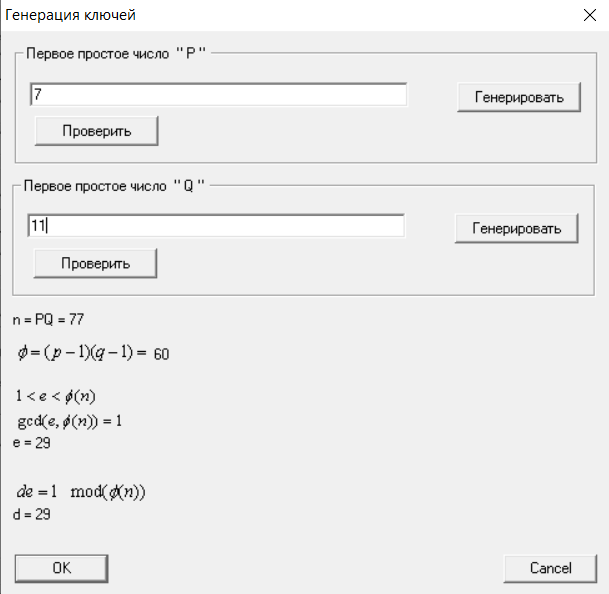
|  |  |
| --- | --- |
|  |  |

3. Рассчитаем степени произвольных больших чисел по модулю прямым методом и быстрым методом.

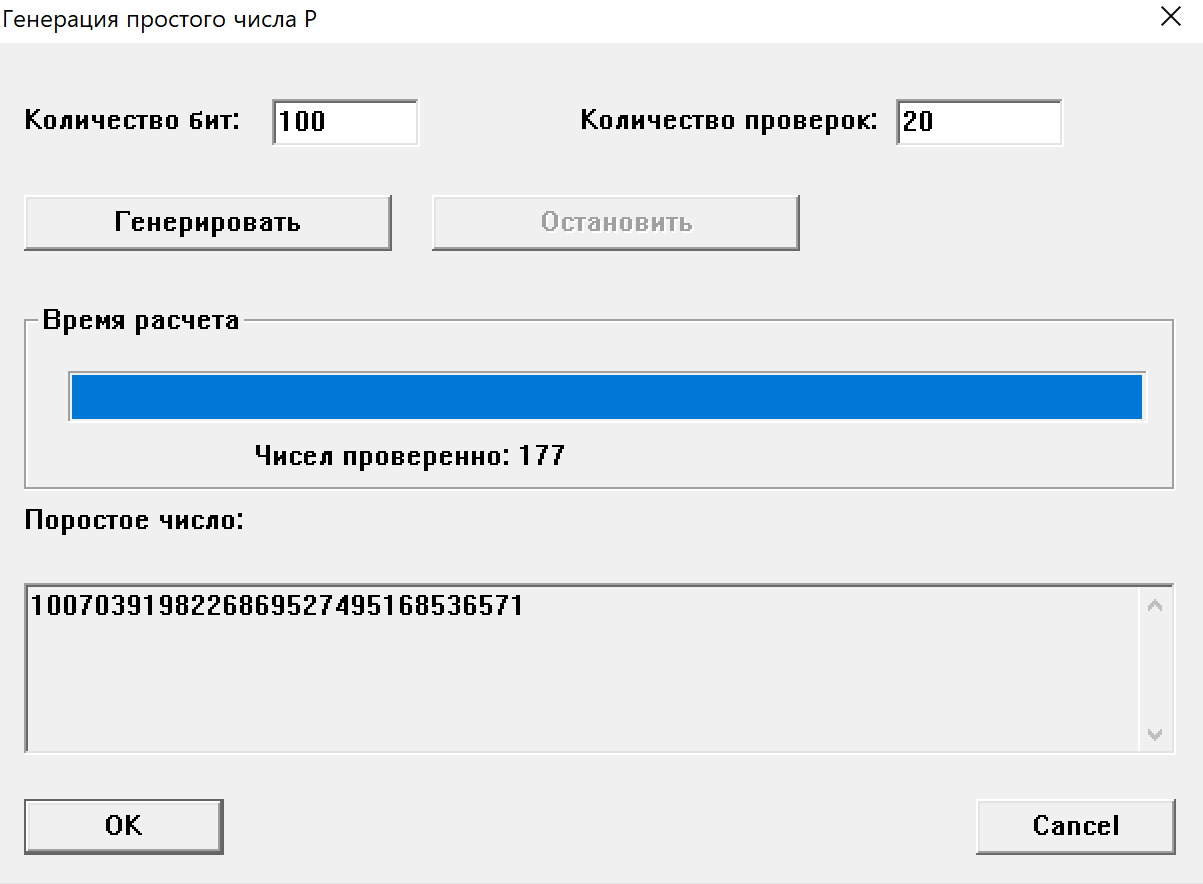
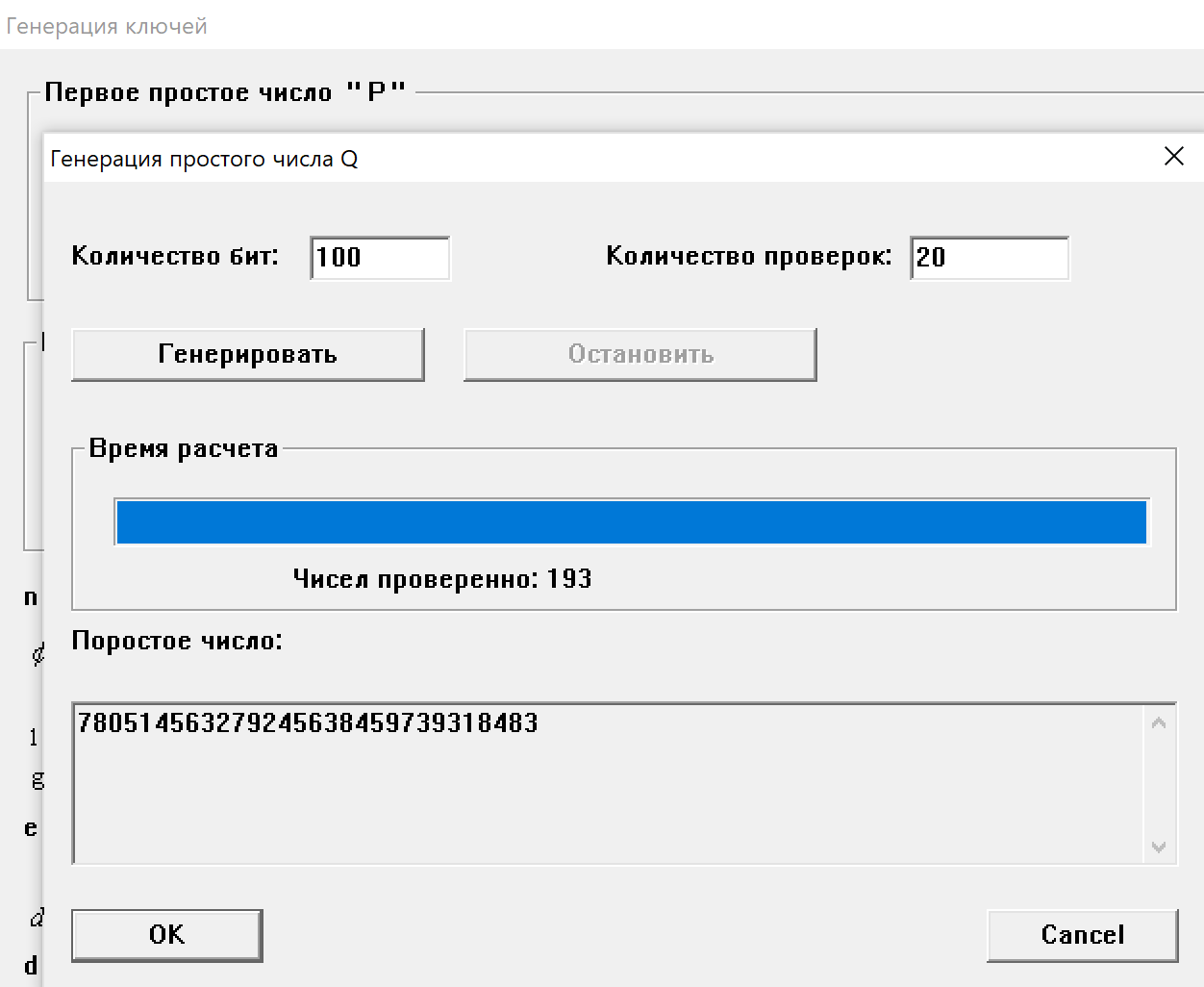
Убедились, что быстрый метод эффективней.

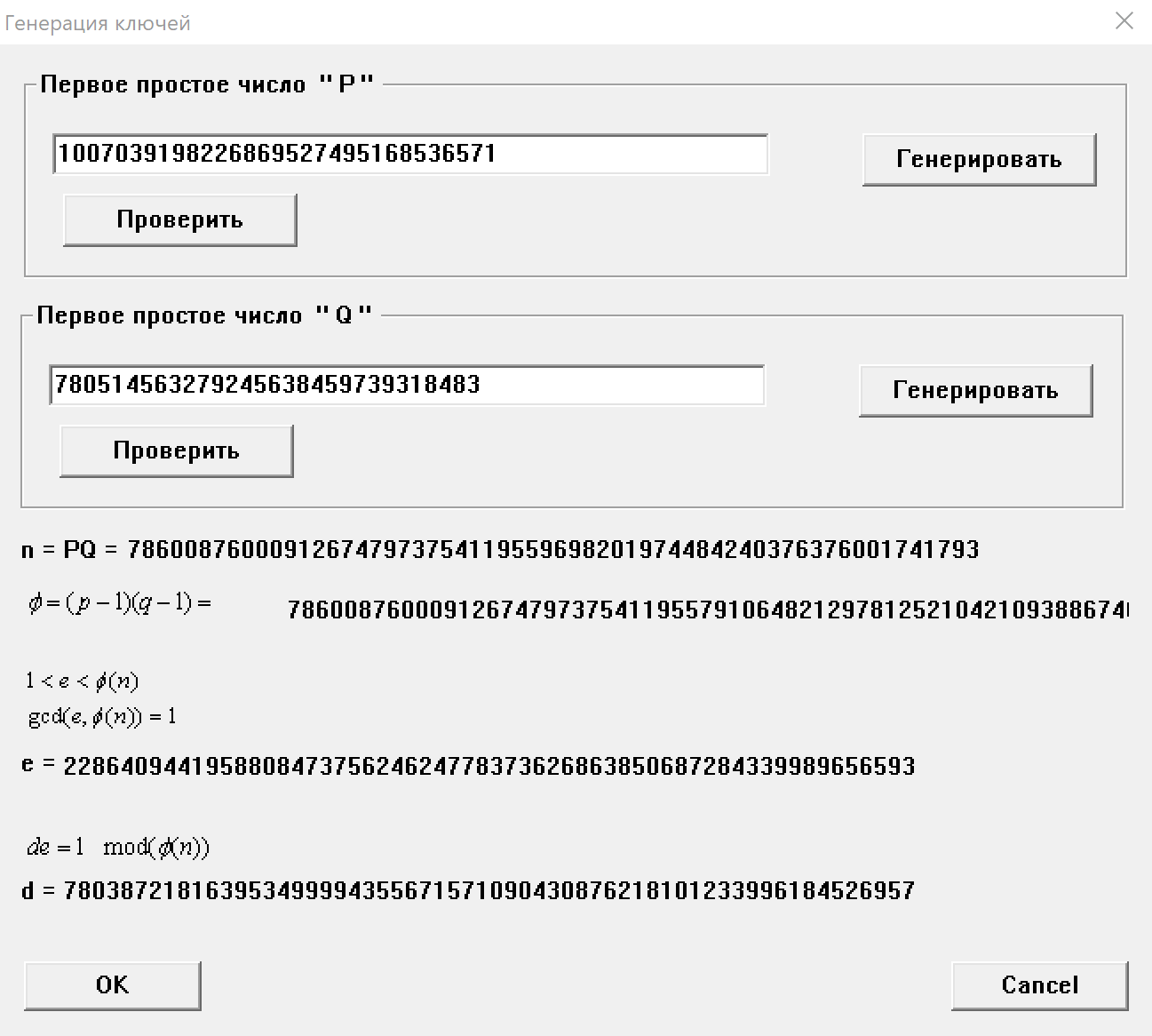
4.Выбрав в главном меню опцию “Генерировать ключи”, зададим малые простые числа p,q и сгенерируем ключи для РША. Проверим правильность вычислений расчетами на бумаге.



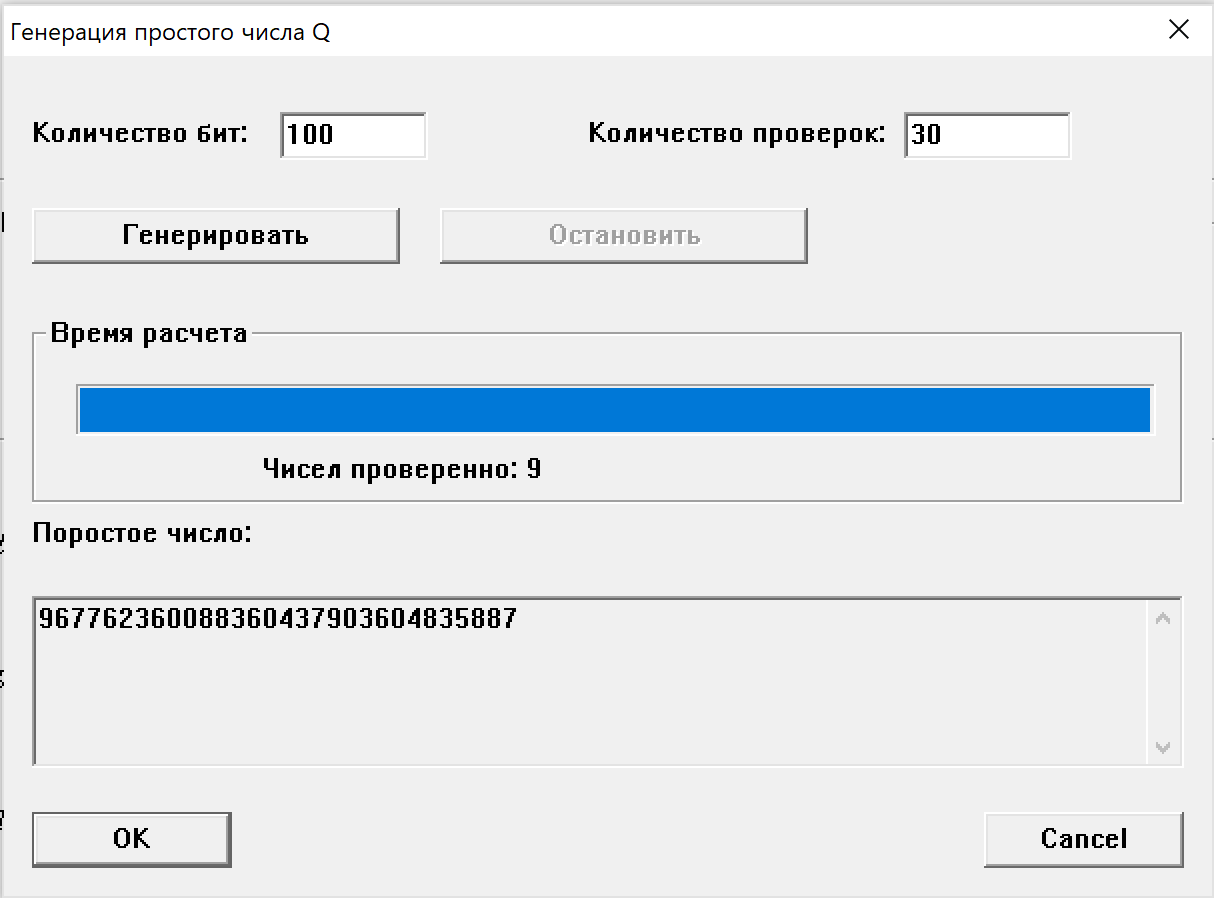
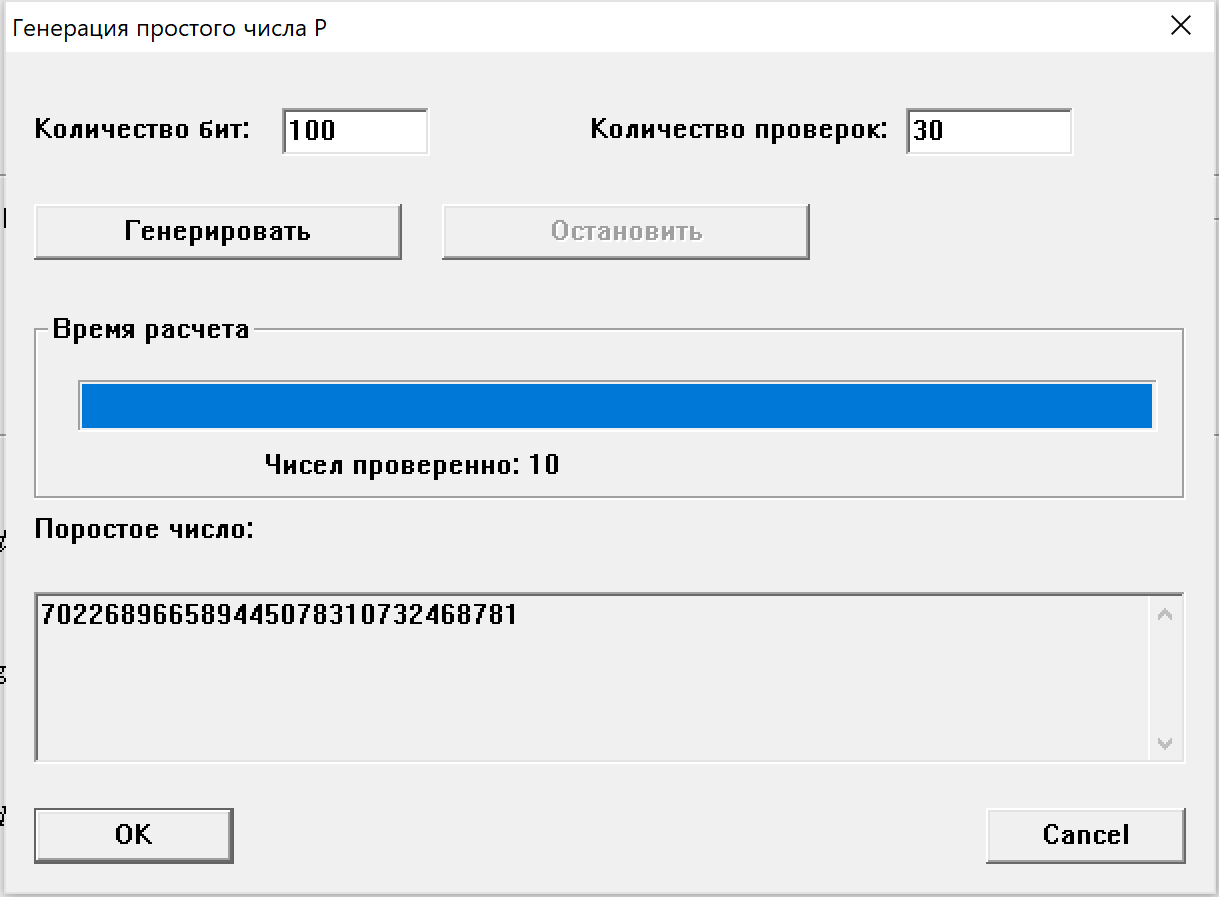
|  |  |
| --- | --- |
|  |  |

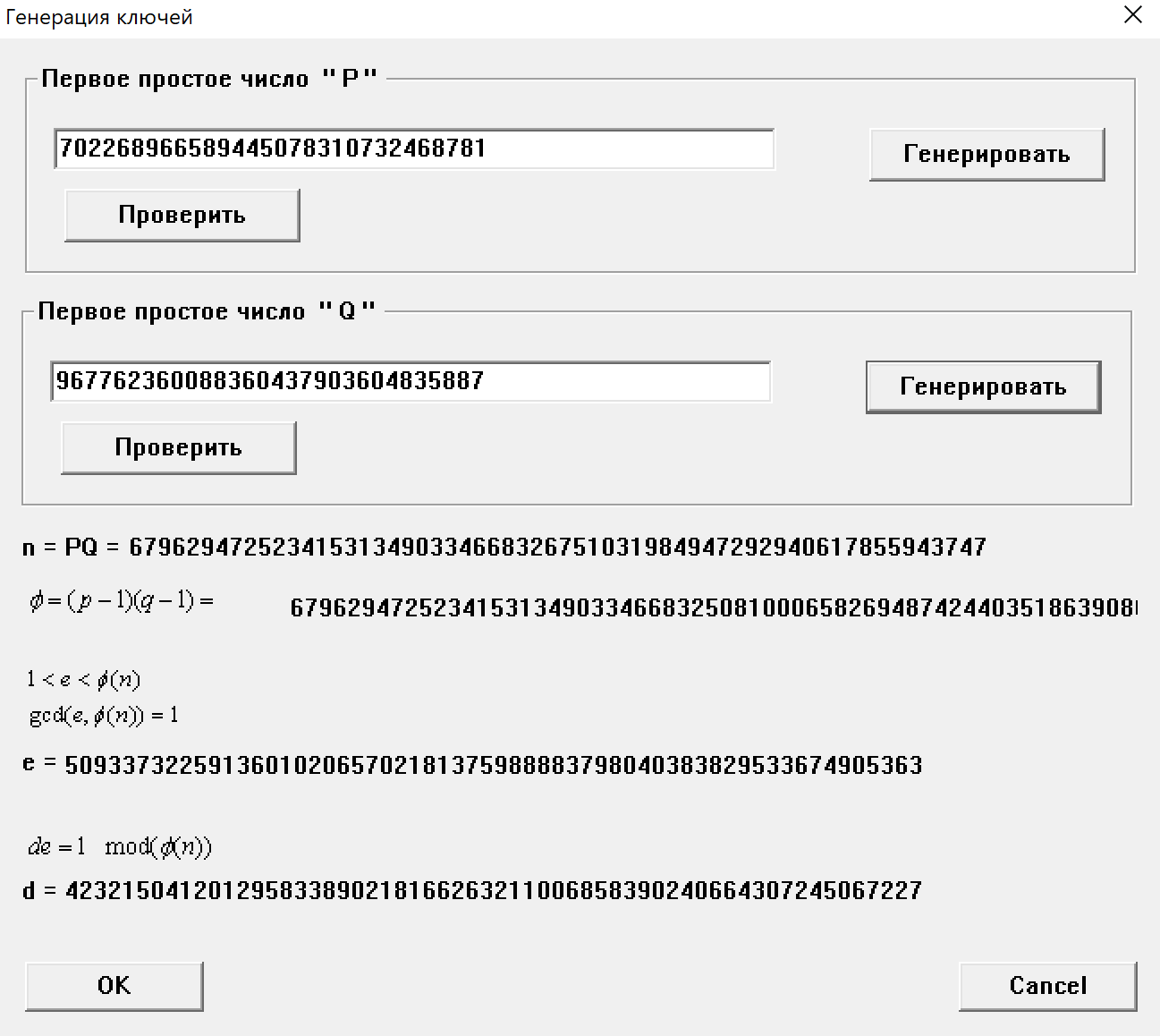
5.Произвести случайное генерирование простых чисел p,q разрядности по 100 бит каждое, задаваясь количеством проверок n=20 и используя опцию “Генерировать” в подменю “Генерация ключей”. Выполнить затем генерацию ключей для системы РША.

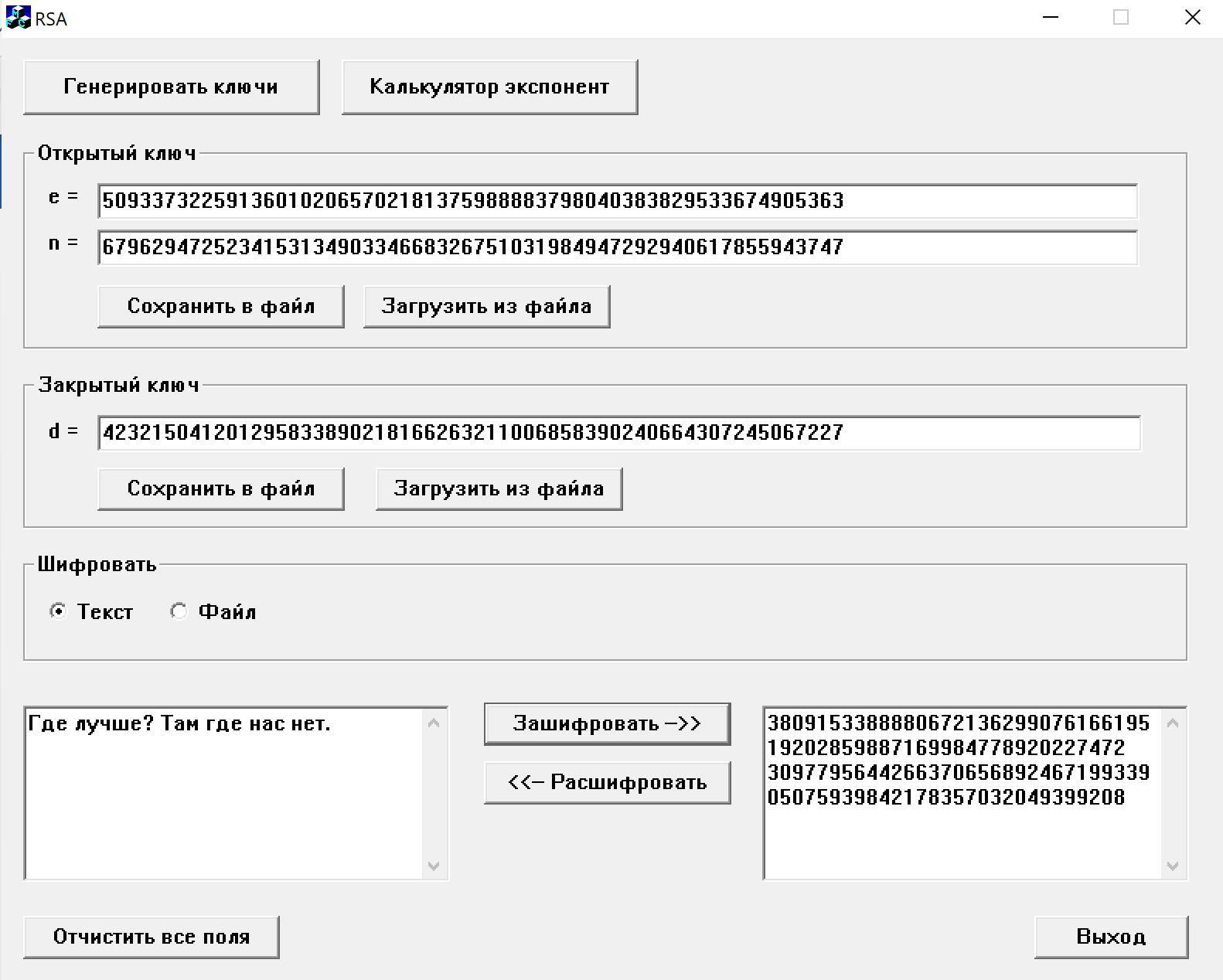


6.Повторим п.5 для n=30. Рассчитаем вероятность ошибочного выполнения тестирования простых чисел для этого и для предыдущего пунктов.

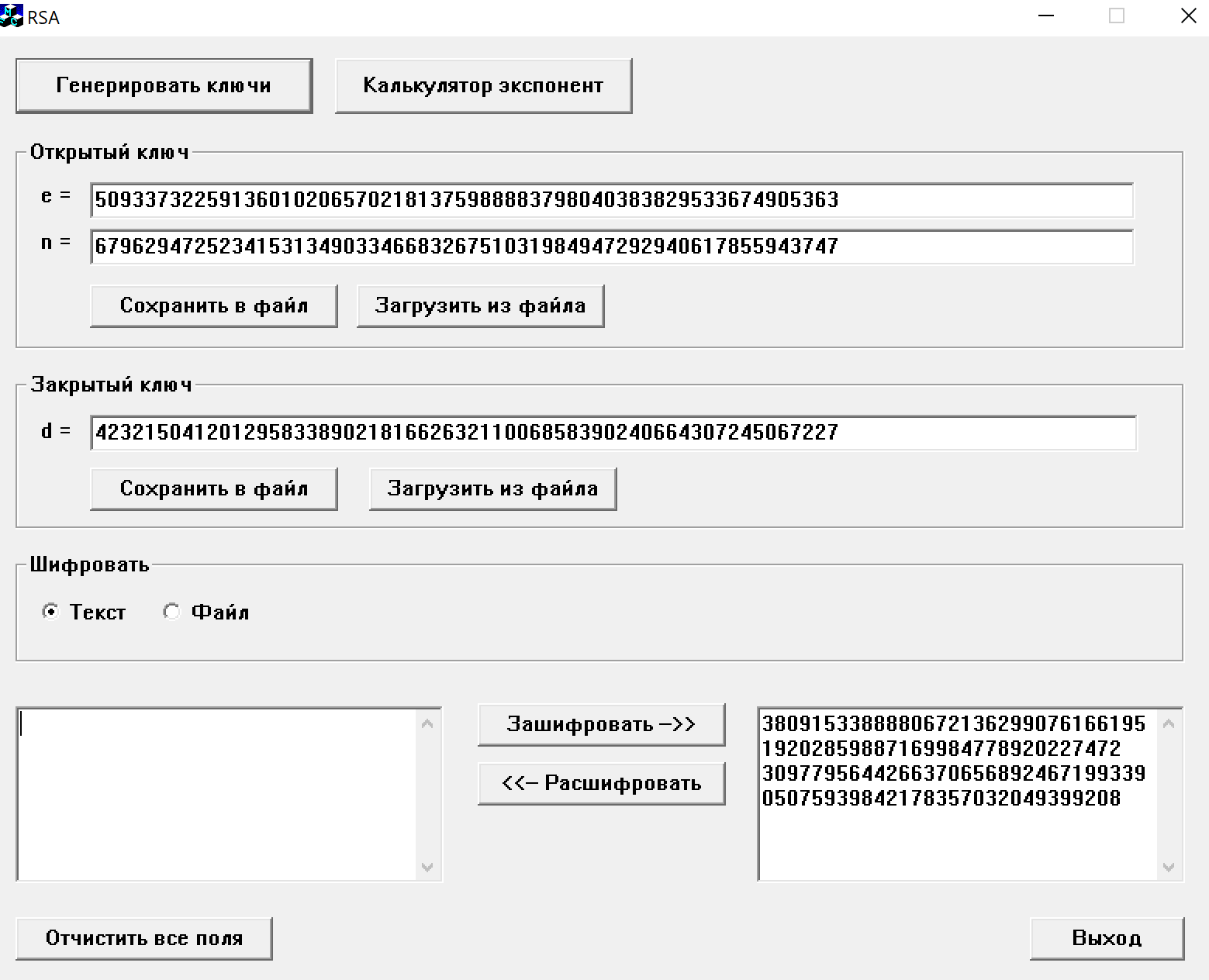
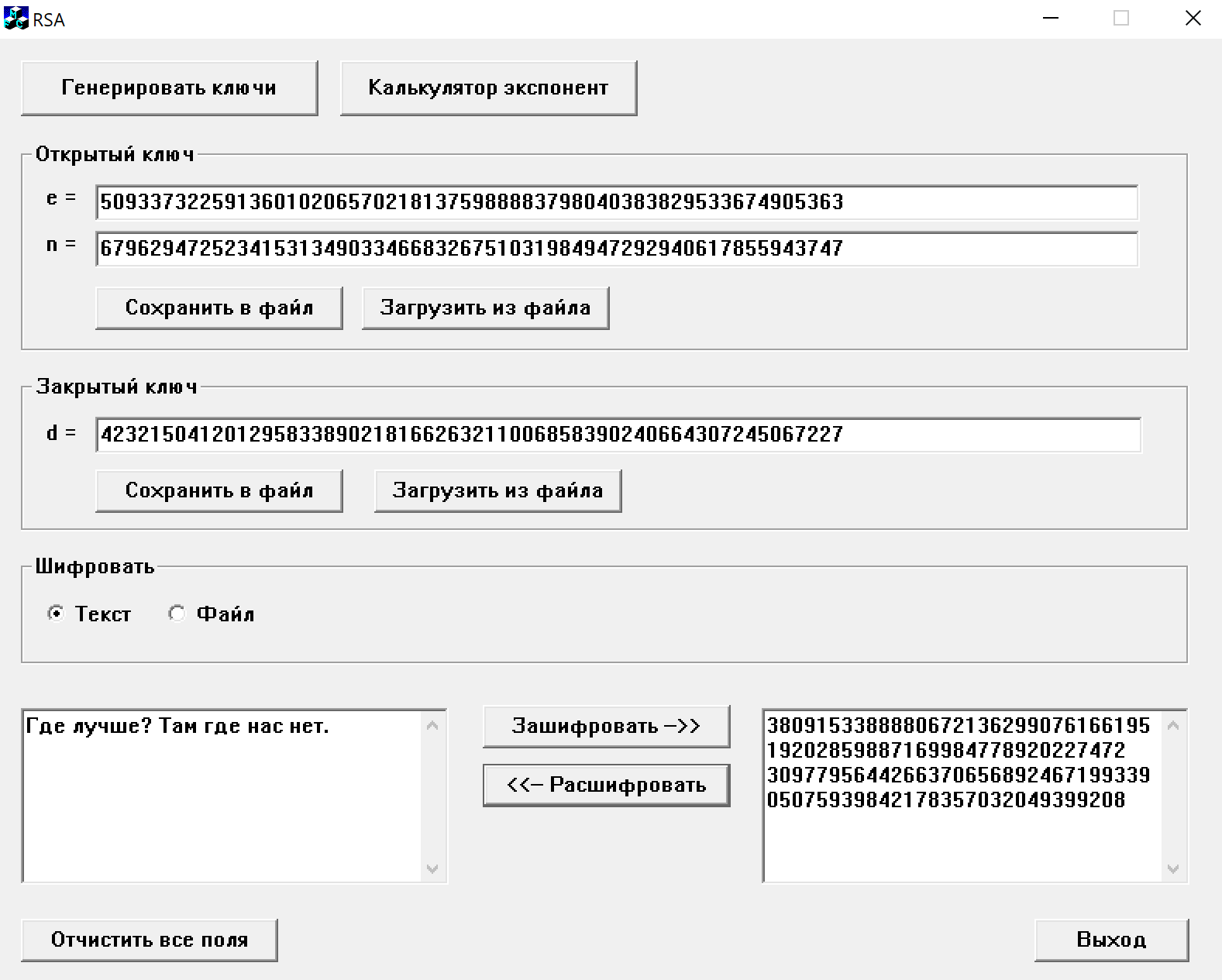




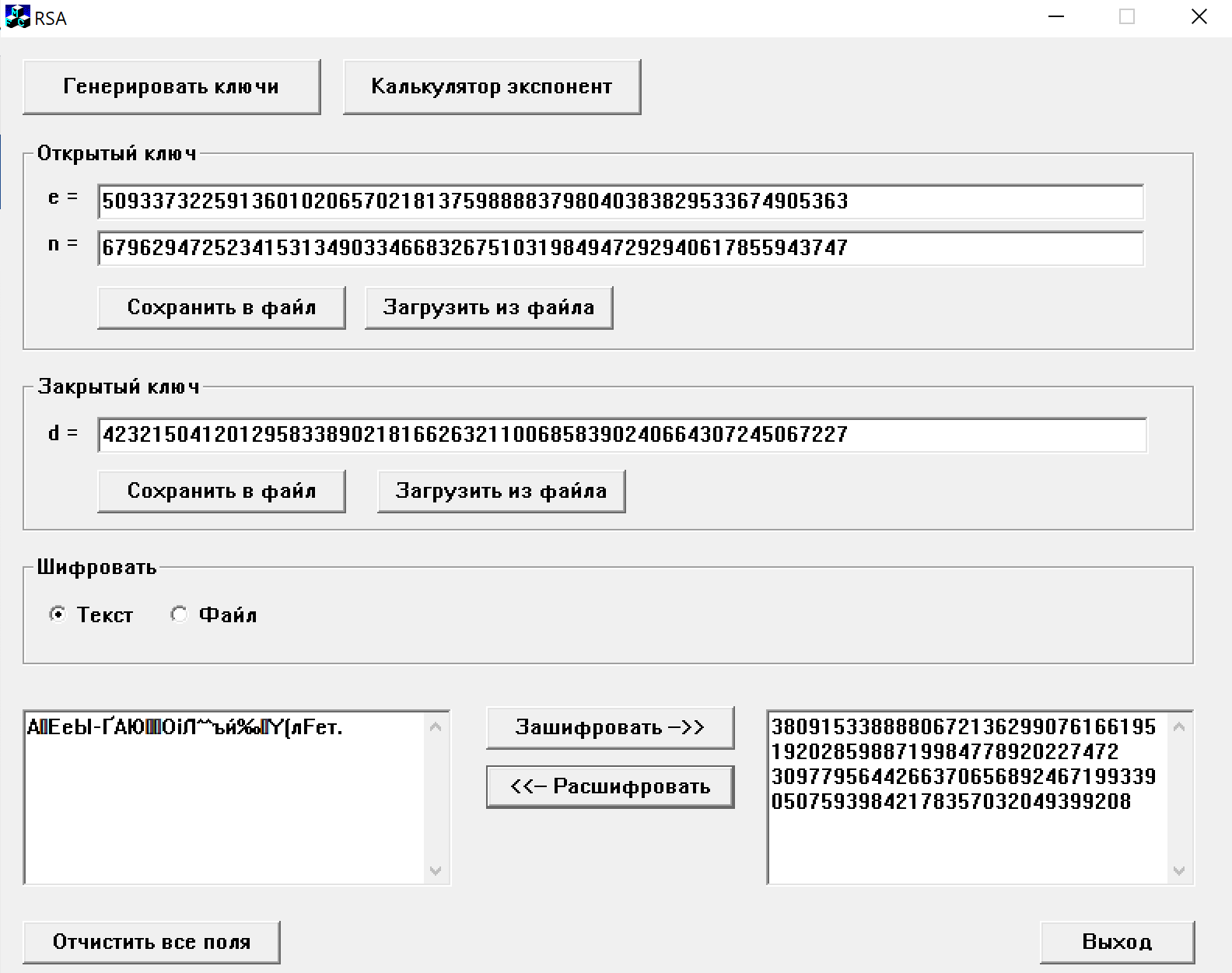
7.Перейдя в основное меню, произведем шифрование короткого смыслового текста на русском или английском языке, введенного в левое поле с использованием опции “Зашифровать”. Пронаблюдаем появление криптограммы в правом поле.



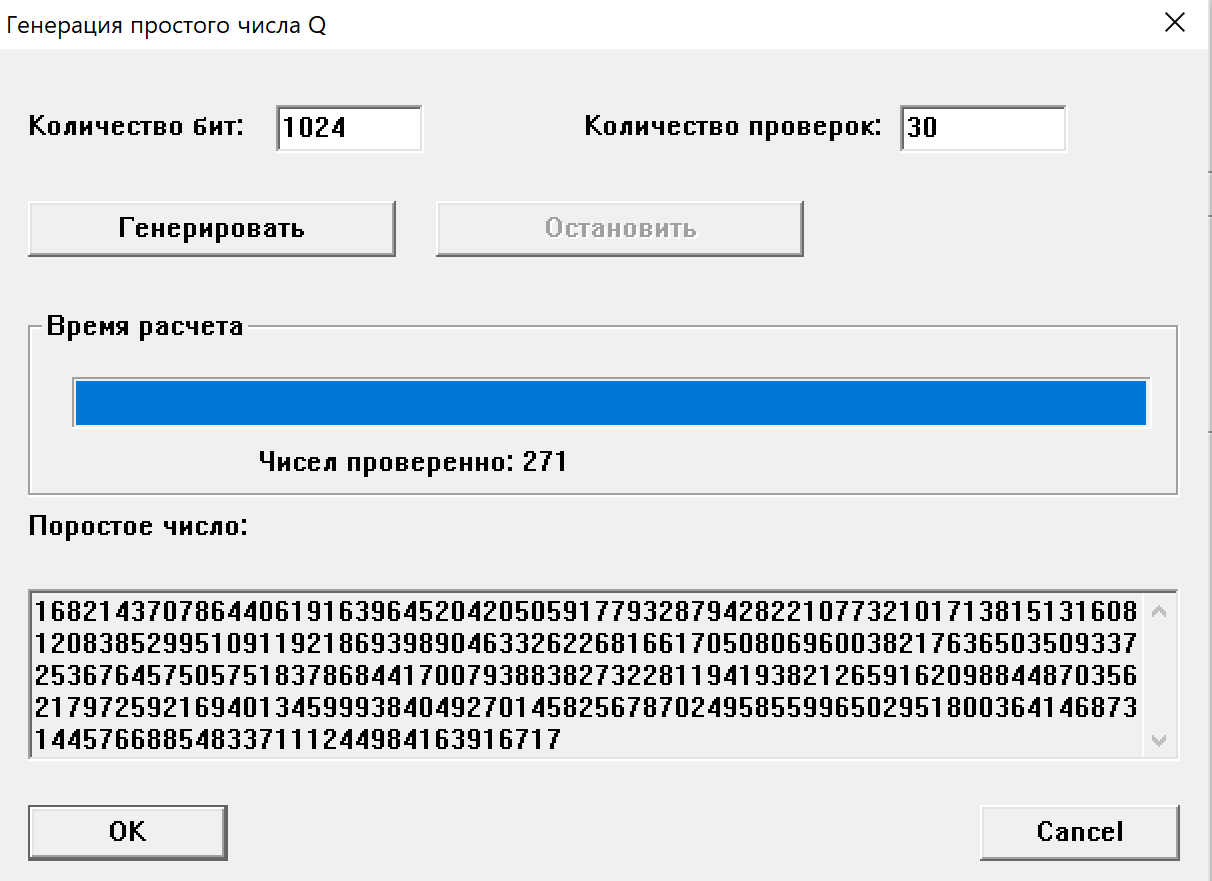
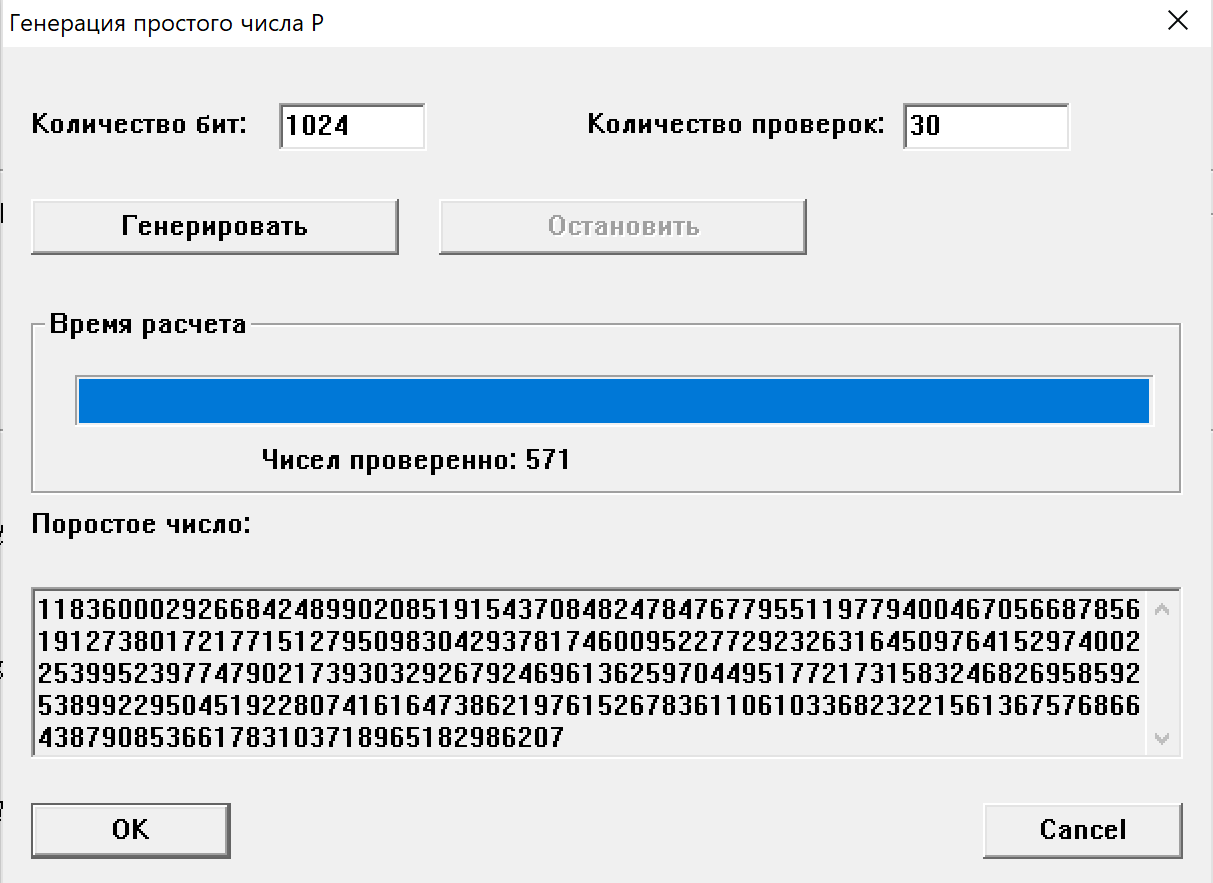
8.Удалим открытый текст в левом поле и произведем дешифрование криптограммы, содержащейся в правом поле, используя опцию “Дешифровать”. Проверим правильность дешифрования.

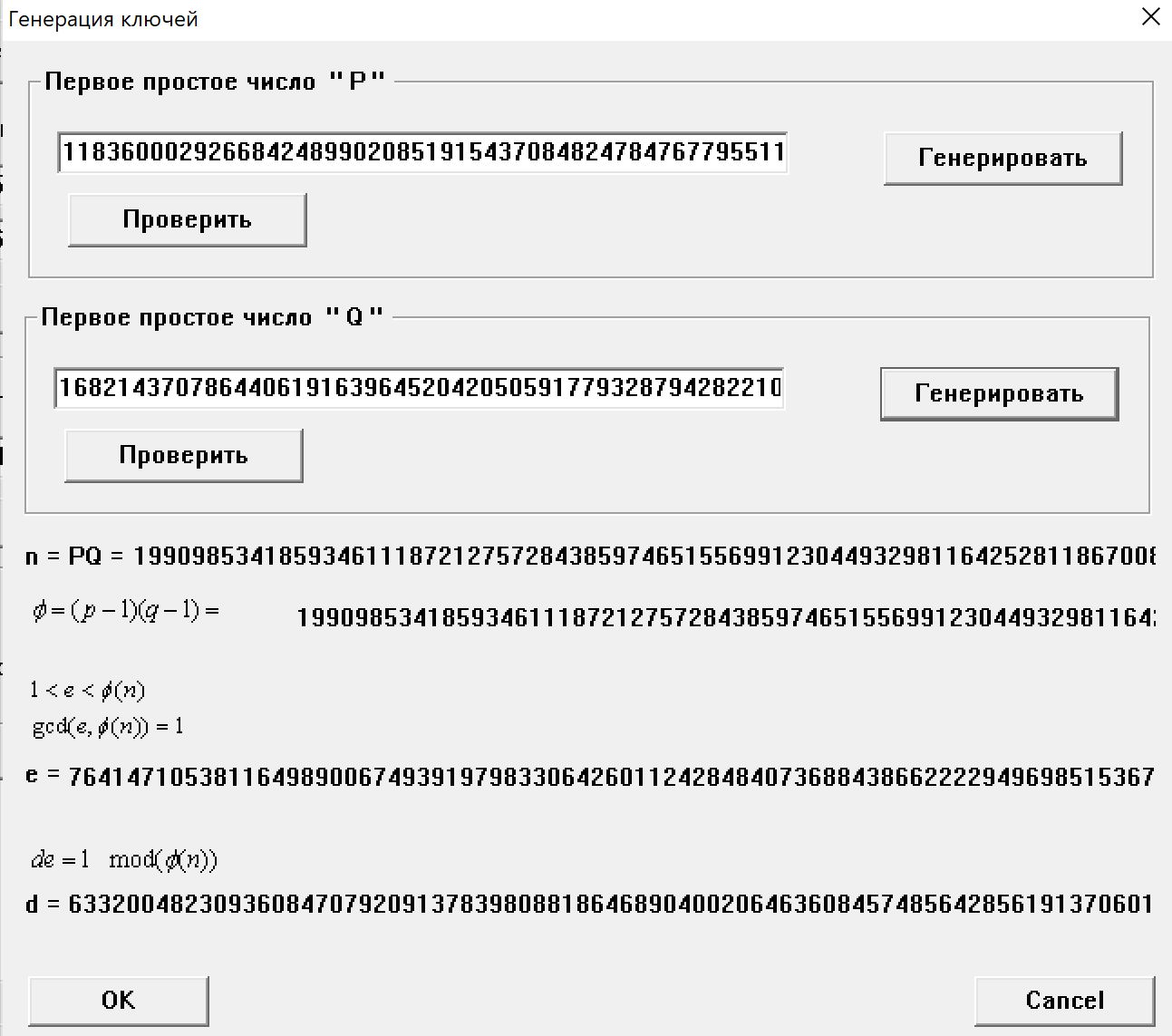
 

9.Повторим п.8 после изменения одной из цифр криптограммы.

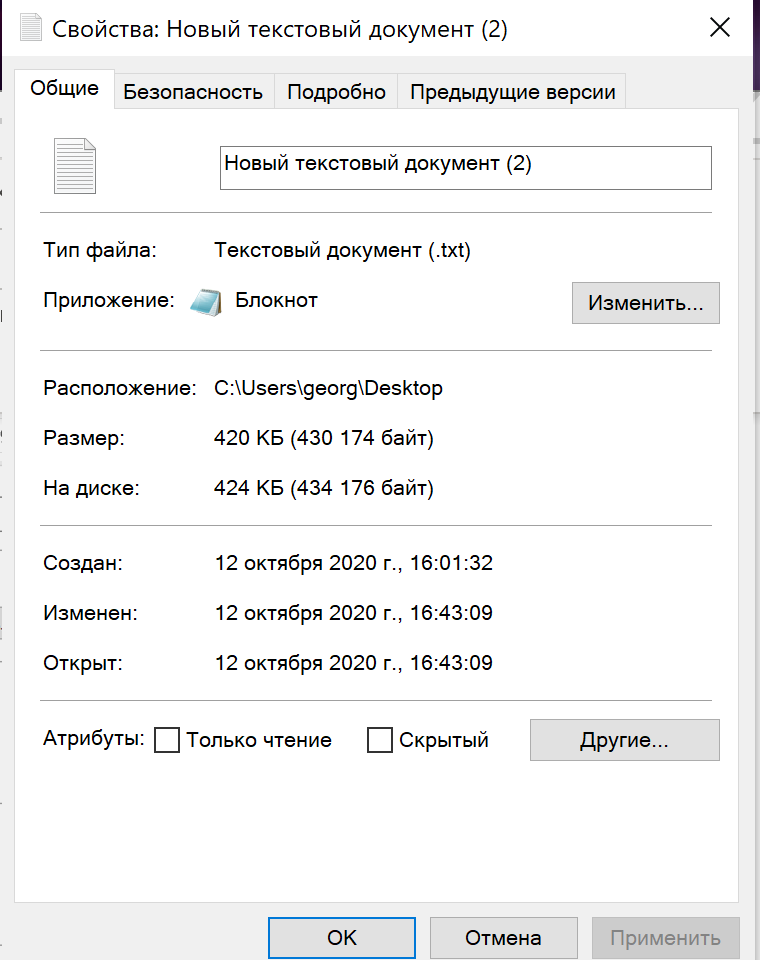
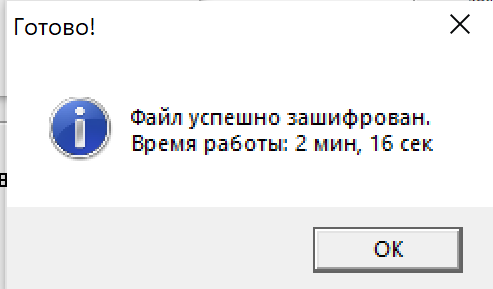


10.Повторить п. 6, выбирая разрядность каждого из простых чисел 1024 бита.

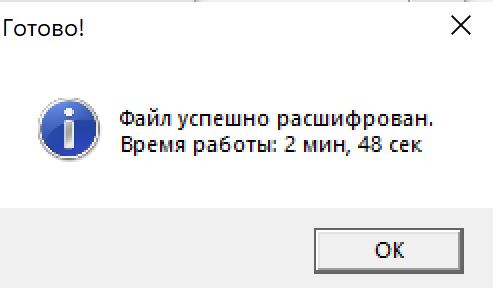




11.Вернувшись в главное меню, произведем шифрование произвольного файла большого объема (не менее 100кбайт), фиксируя при этом время шифрования.

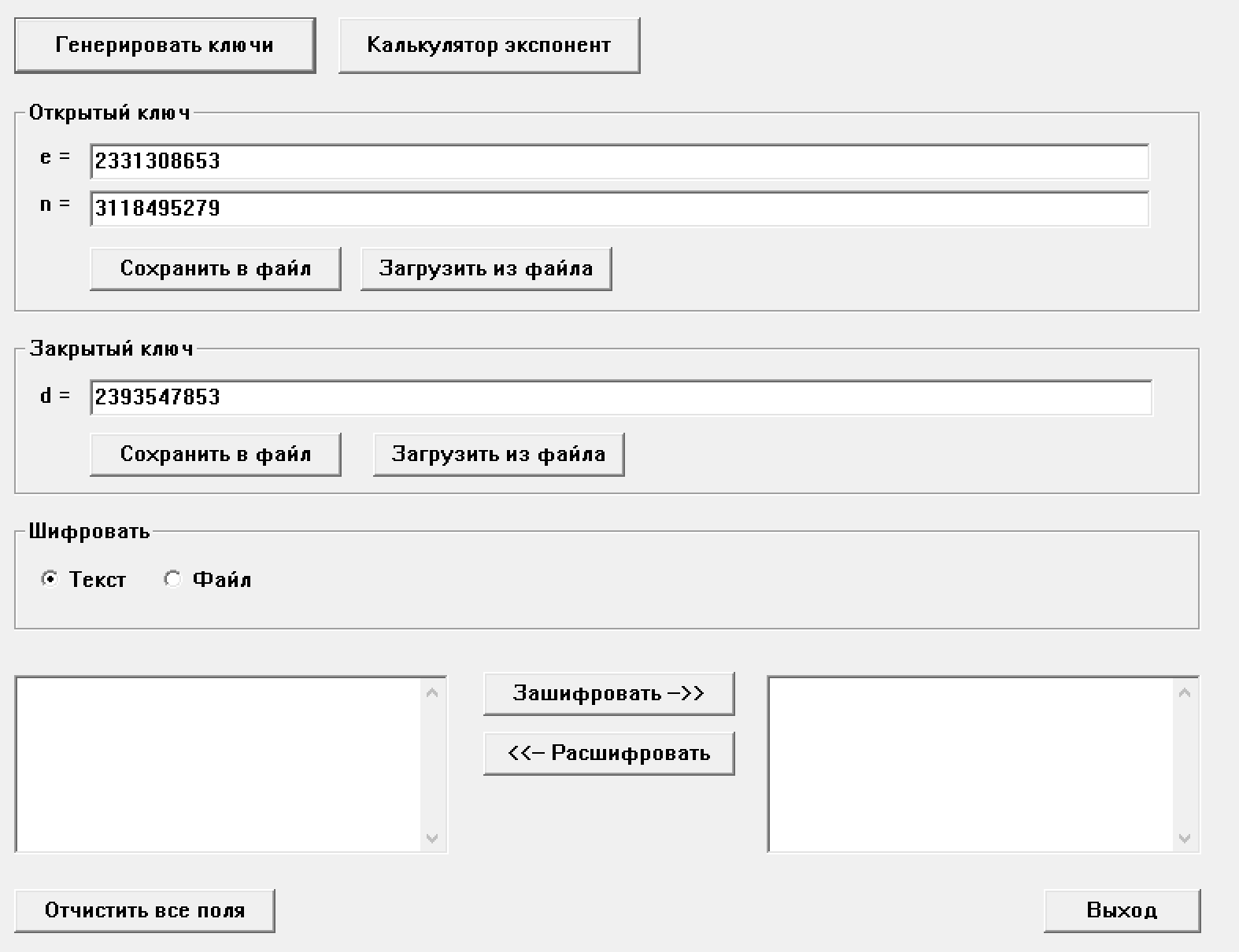
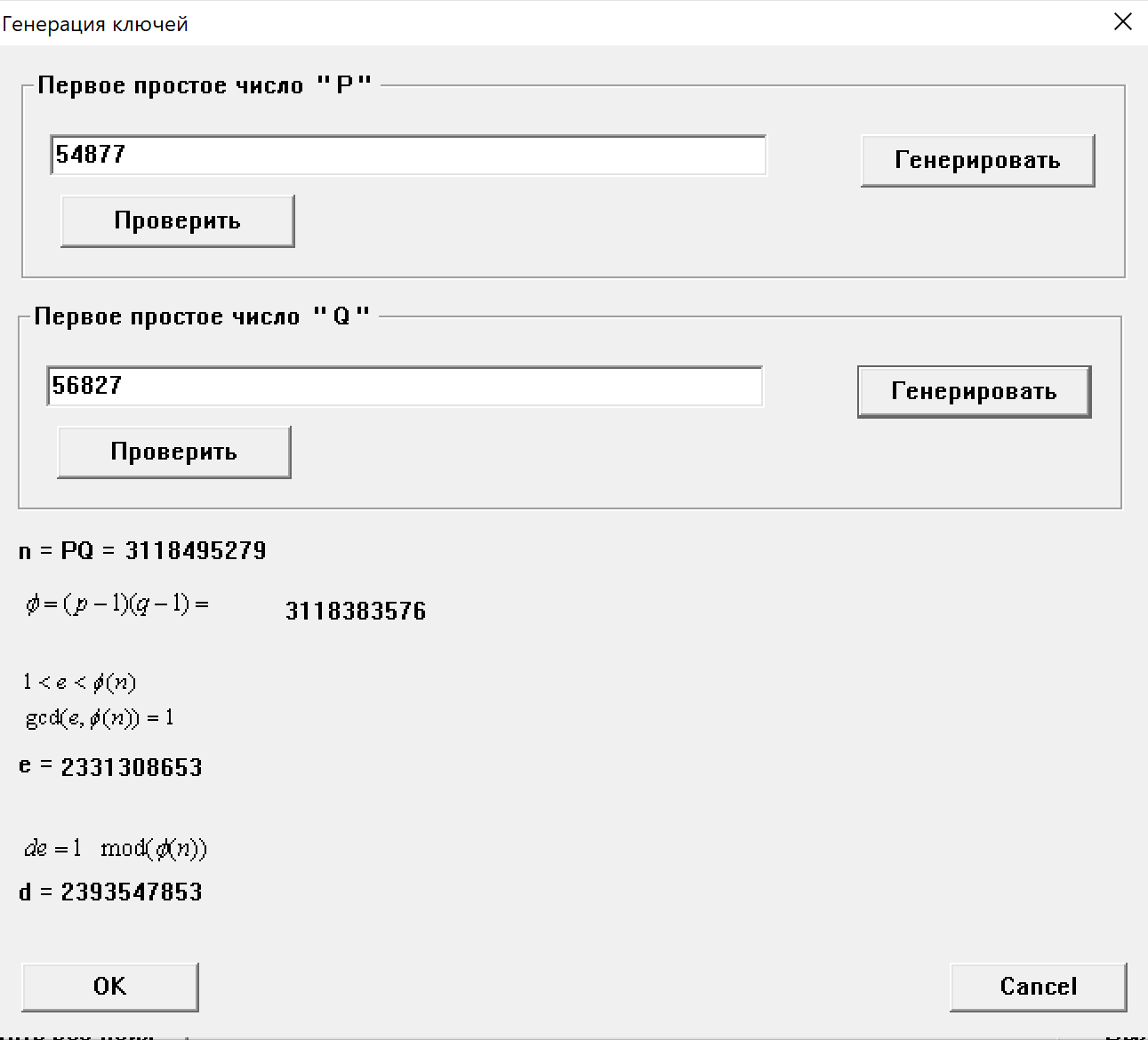
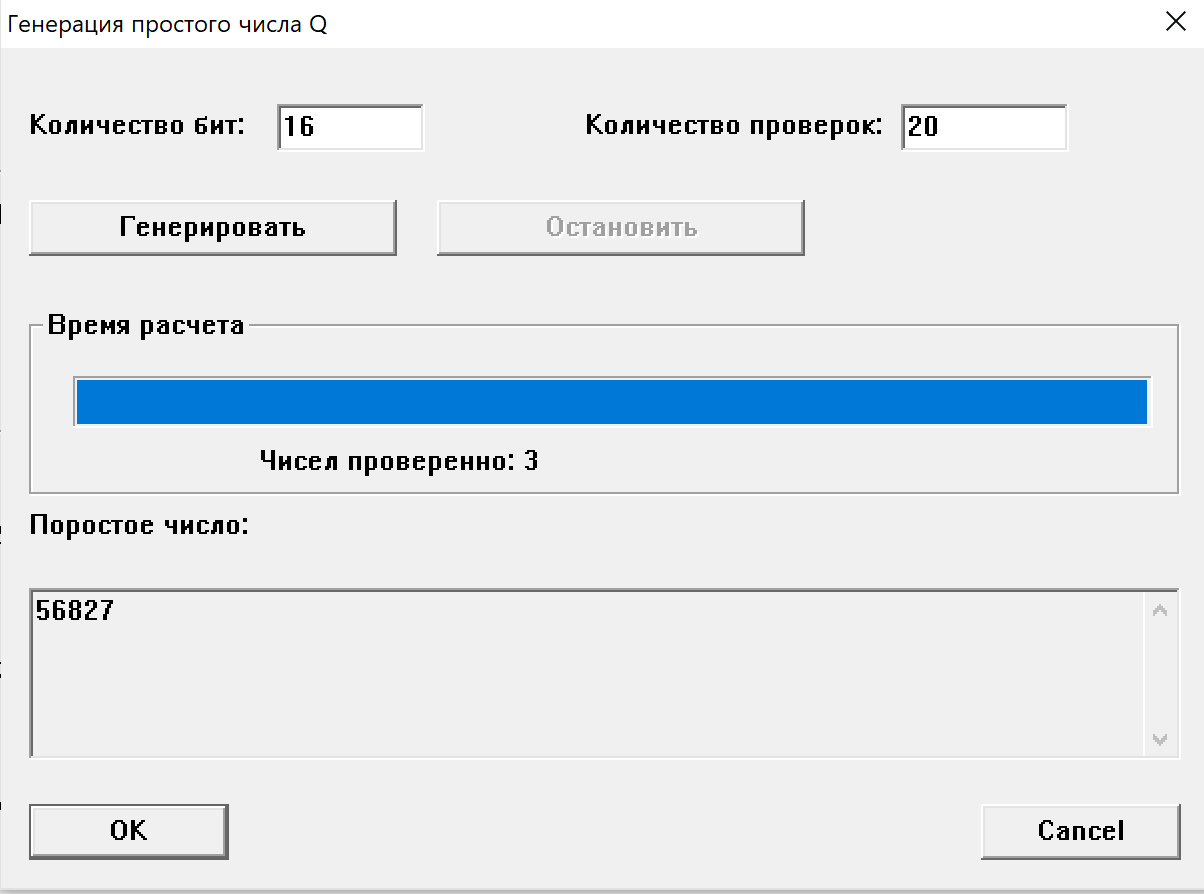
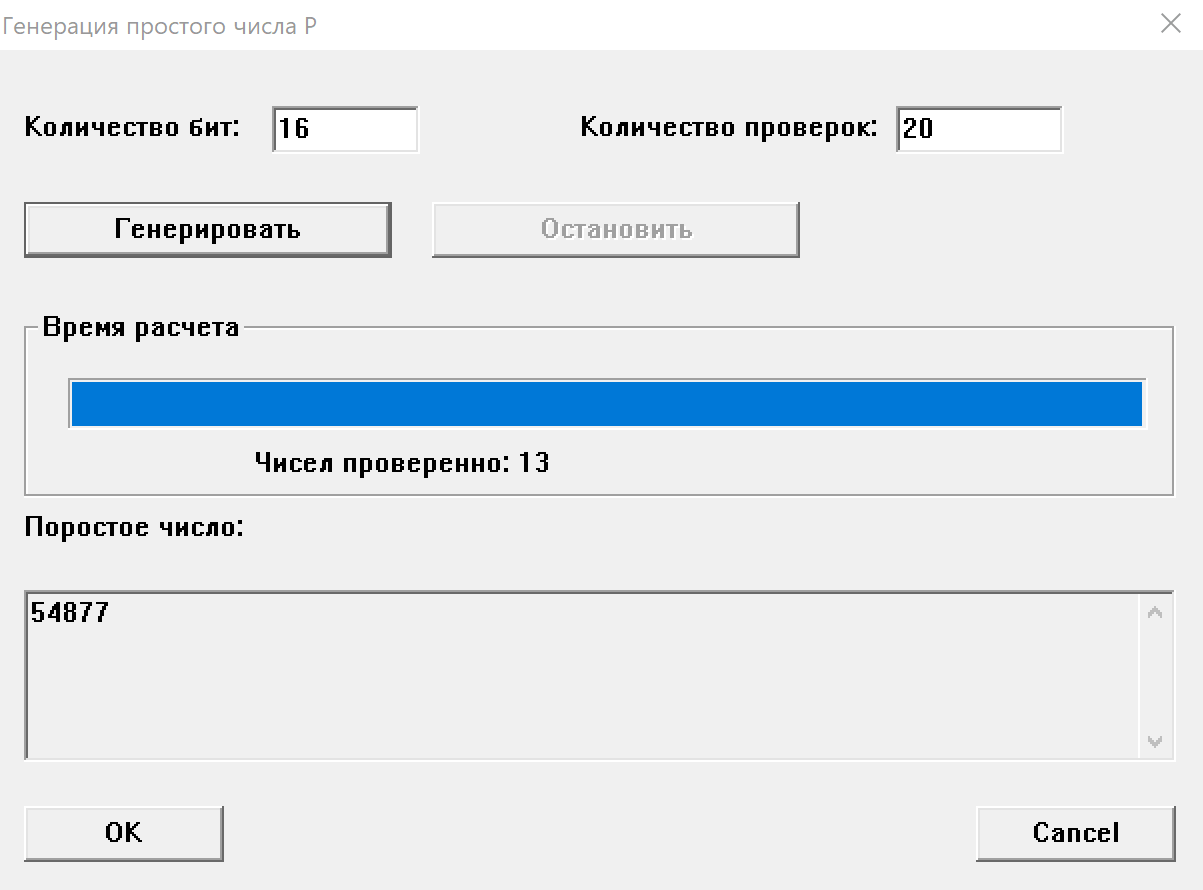
 

12.Произведем дешифрование криптограммы, полученной в п.11 и помещенной в специальный файл, фиксируя при этом время дешифрования.

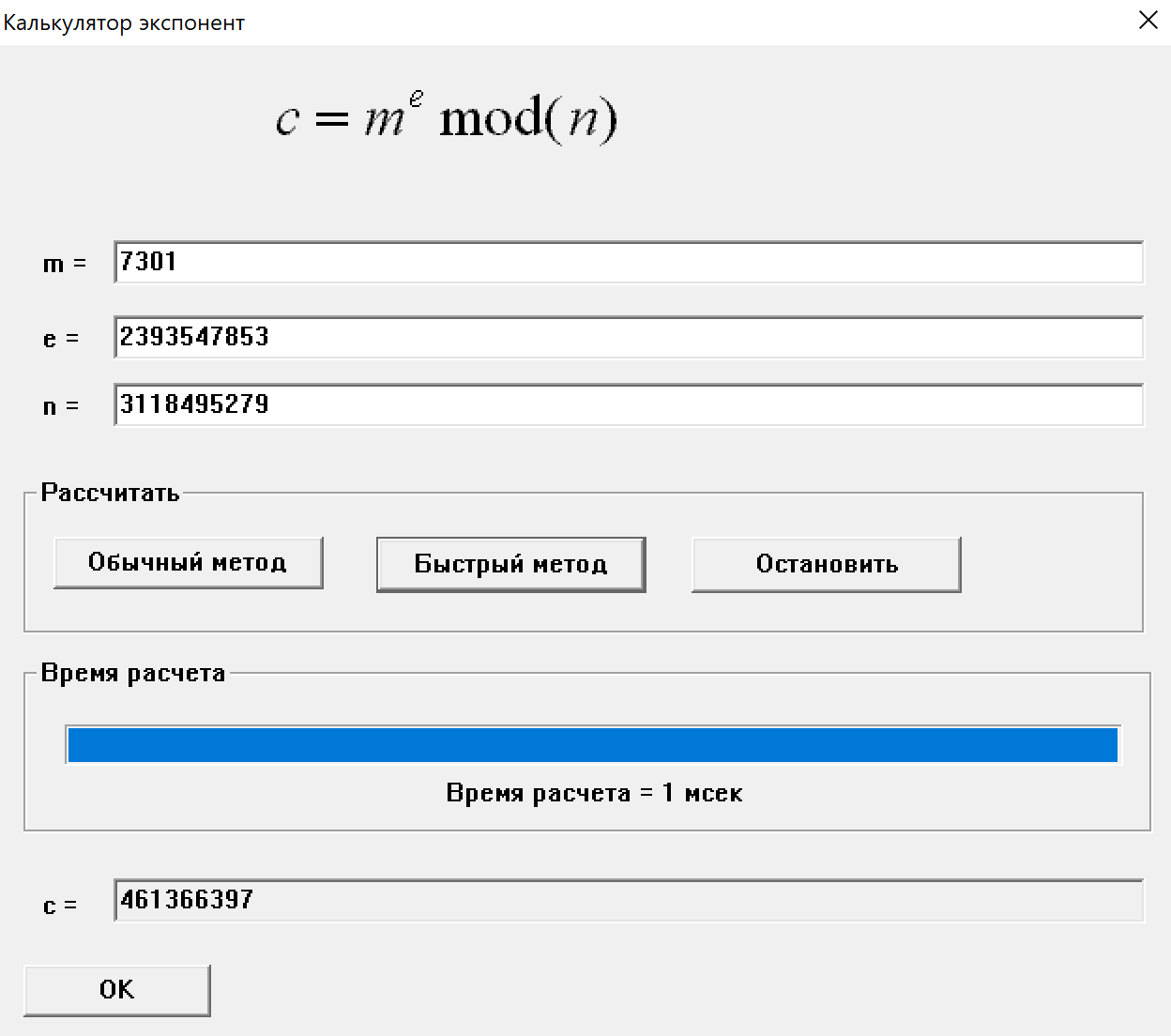


13.По данным п.11,12 рассчитать скорость шифрования и дешифрования при данной простейшей программной реализации безопасной криптосистемы РША.

14. Повторим п. 5, выбирая разрядность каждого из простых чисел 16 бит.

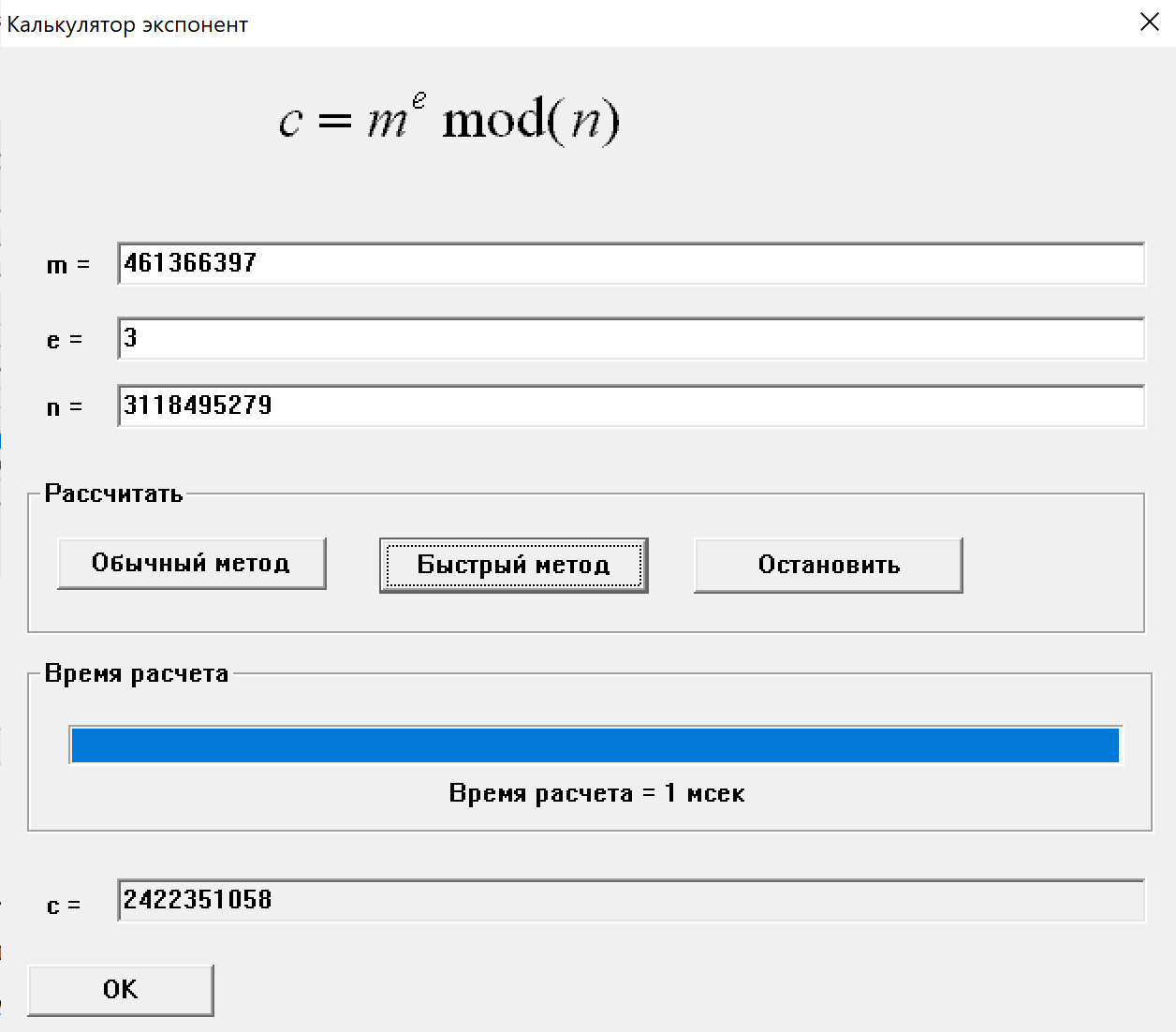
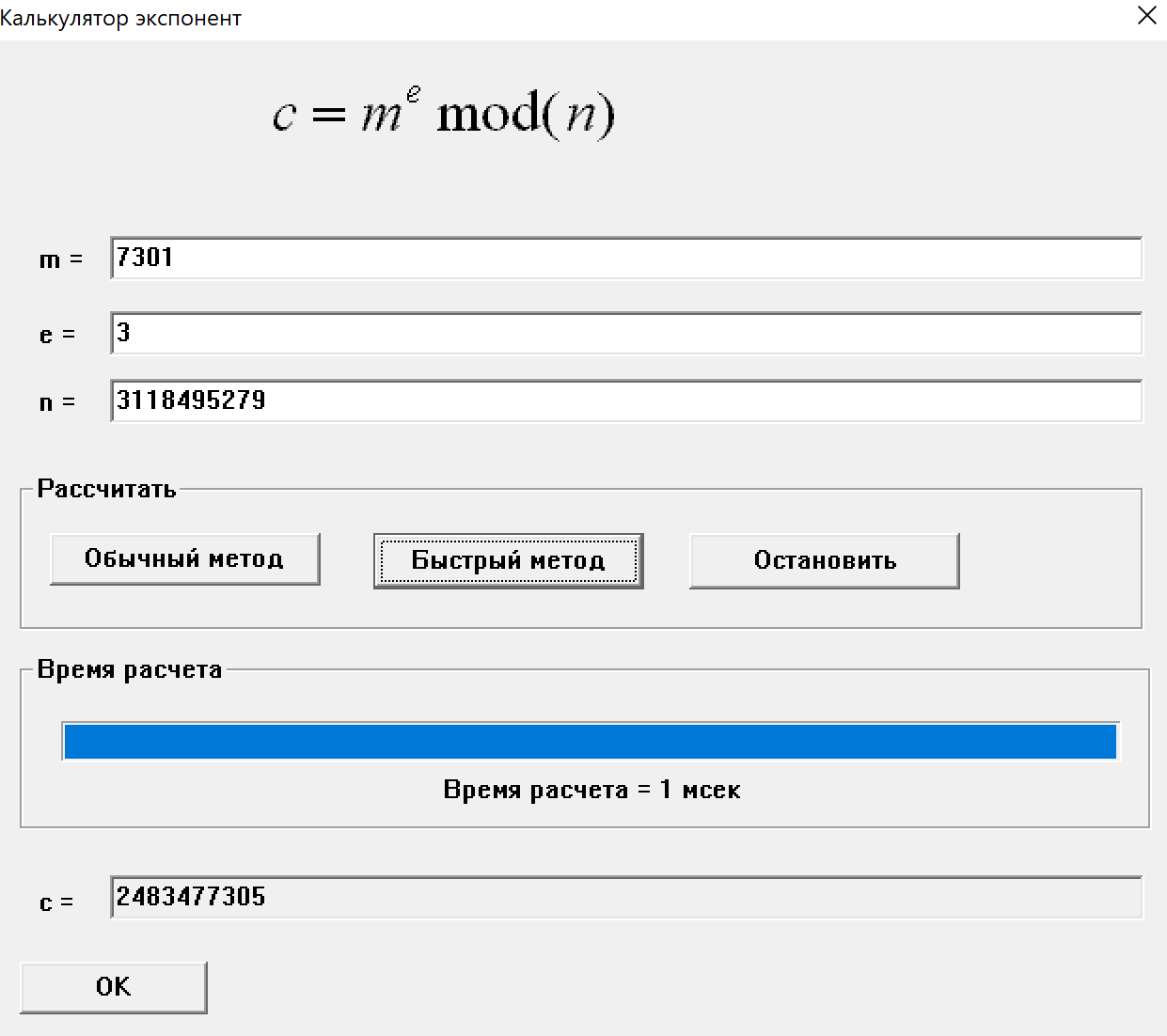


15. Используя подпрограмму «Калькулятор экспонент», сгенерируем цифровую подпись открытого текста m в виде ГГNN (где ГГ- номер группы, NN - ваш номер по журналу). без использования хэш-функции (аналогично шифрованию, но с закрытым ключом) . Подписью будет пара чисел (m,s).



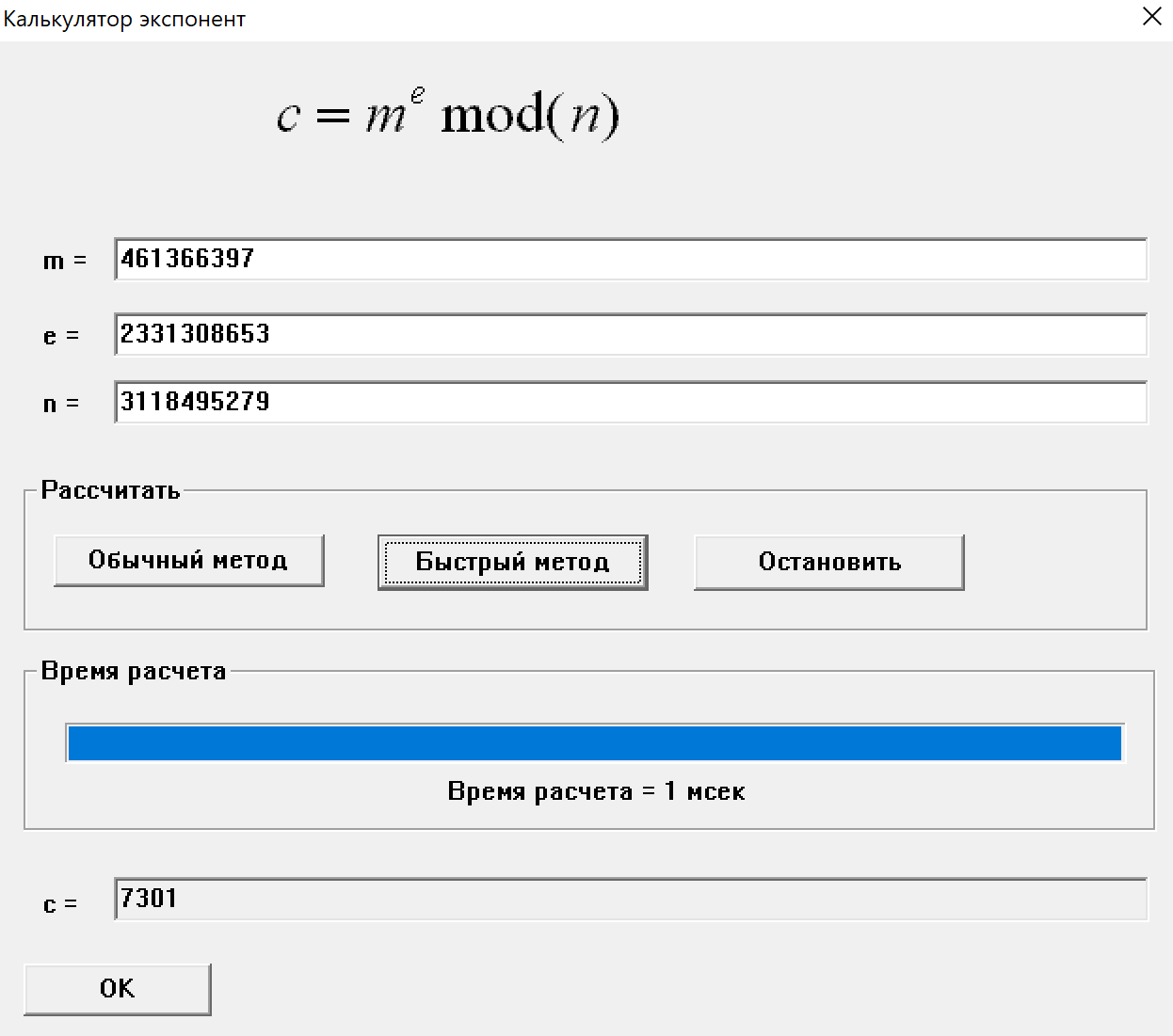
Подпись (7301, 3118495279)

16. Создадим подделку цифровой подписи путем возведения m и s в степень с произвольным показателем l: , . (Используем ту же программу вычисления экспонент, l = 3).

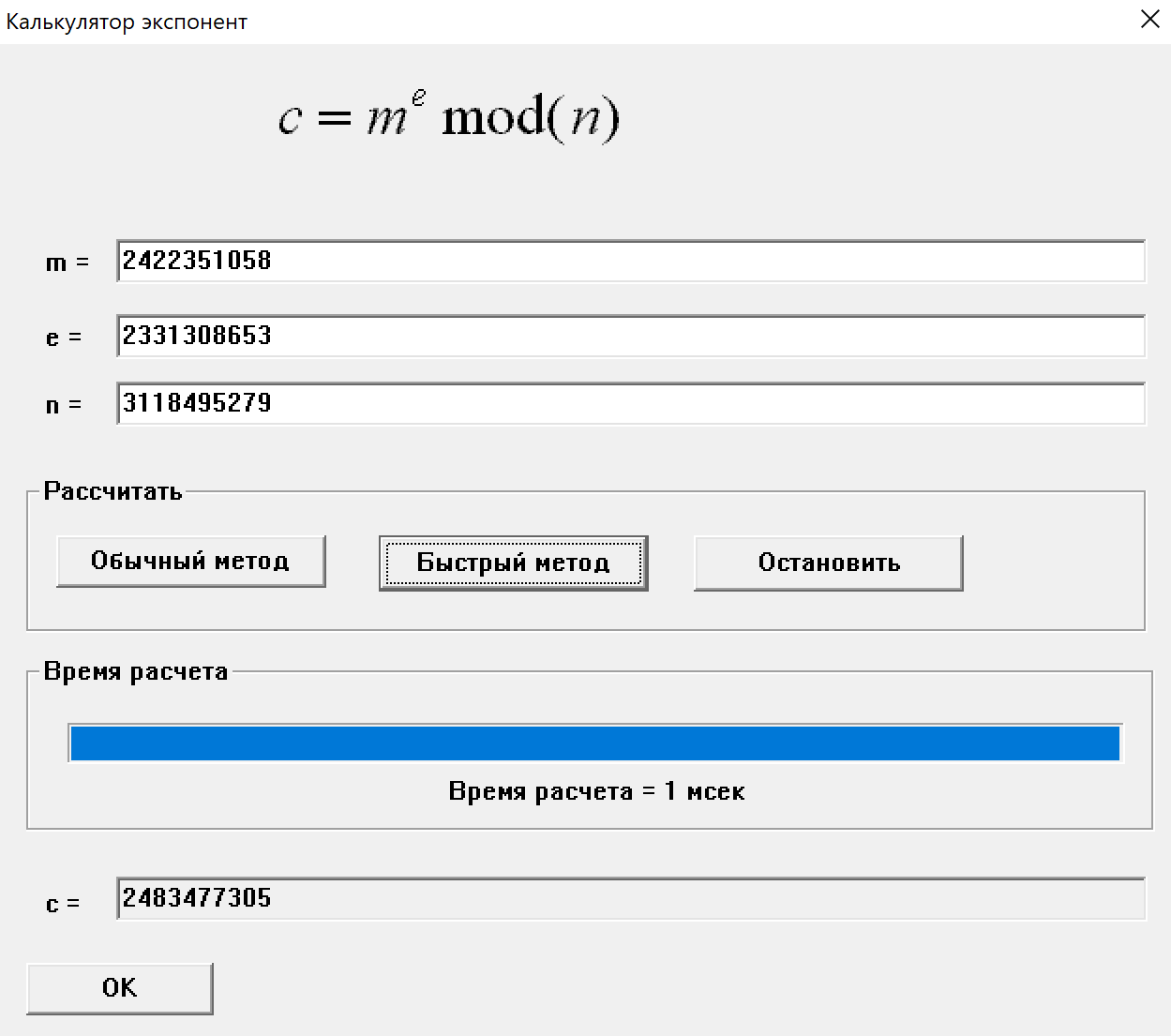


17. Проведем верификацию цифровой подписи в обоих случаях:

* найдем *m1=se(mod n)* ипроверим равенство *m1=m*



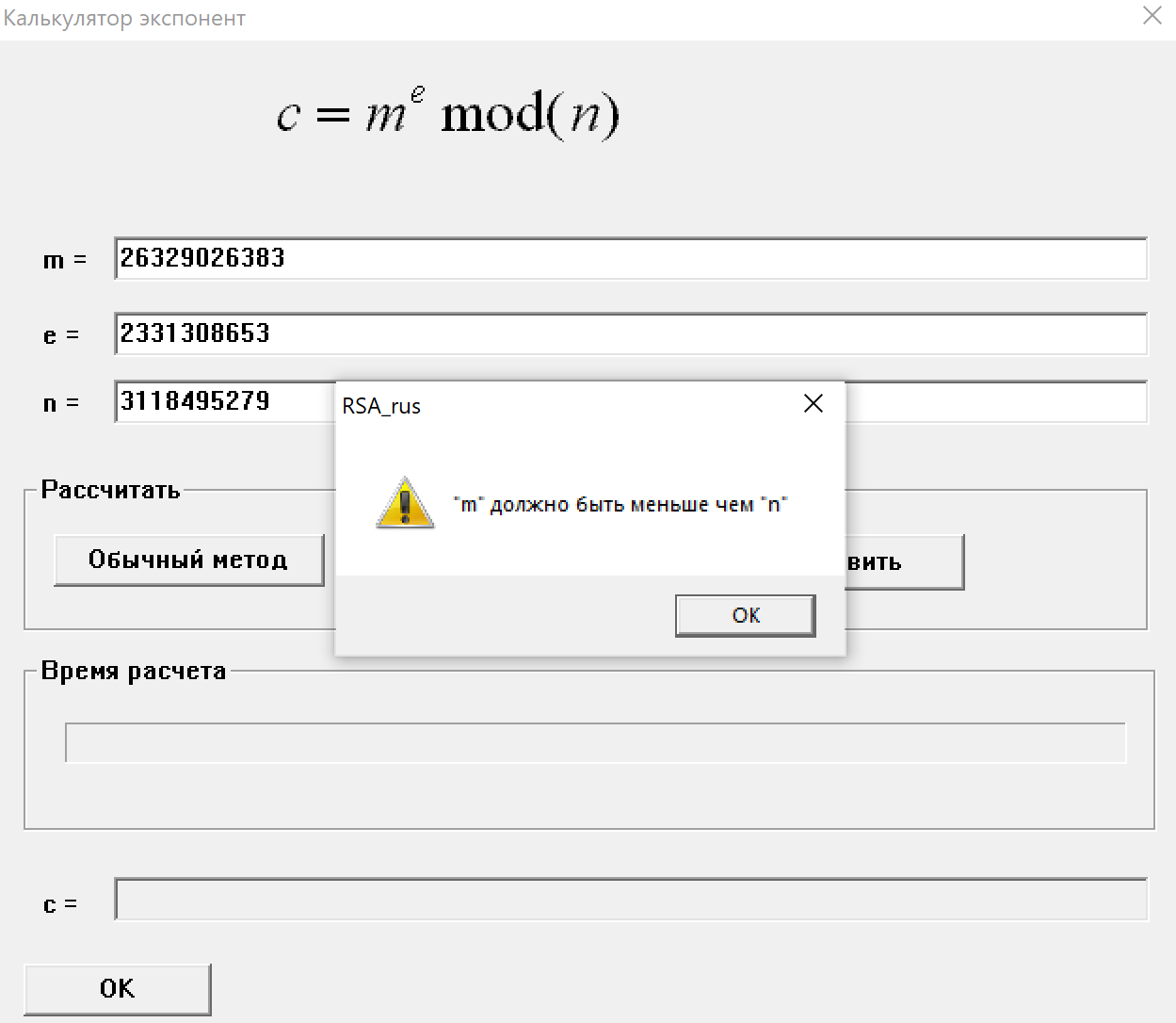
* найдем *m2=s’e(mod n)* ипроверим равенство *m2=m’*.

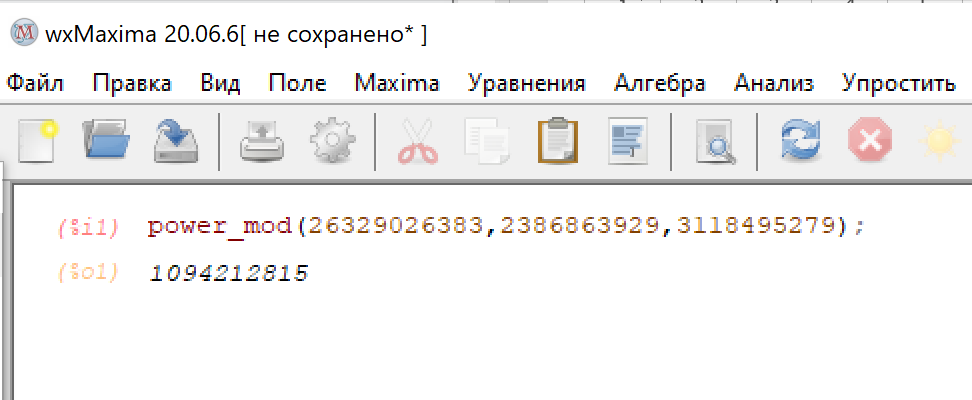


Безопасность и целостность цифровой подписи открытого текста без применения хеширования, судя по проведенным опытам, не обеспечивается.

18. Получим хэш-образ открытого текста h(m), используя следующий упрощенный алгоритм: сложим номера всех символов сообщения. Результат умножим на d и приведем по modn.

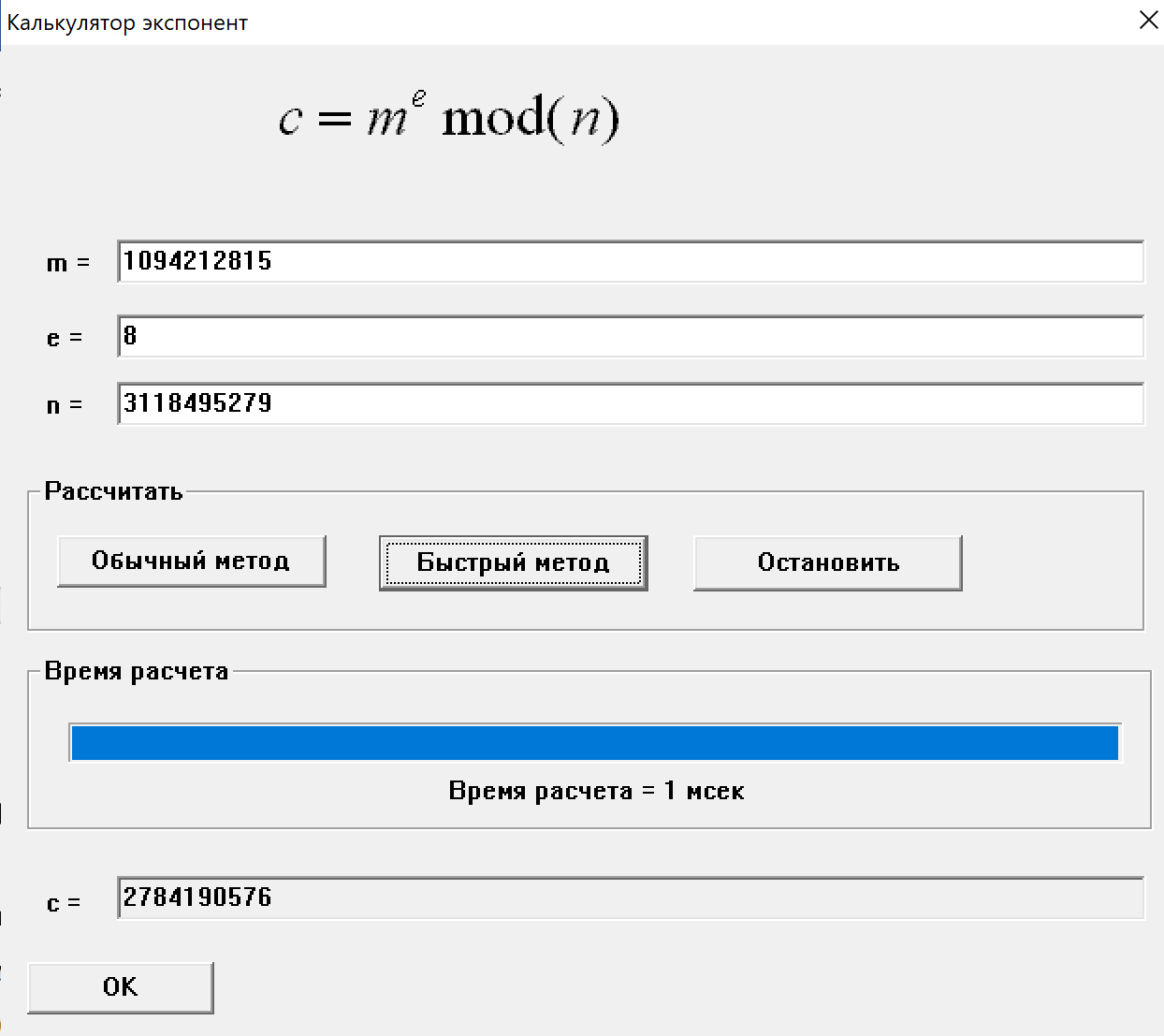
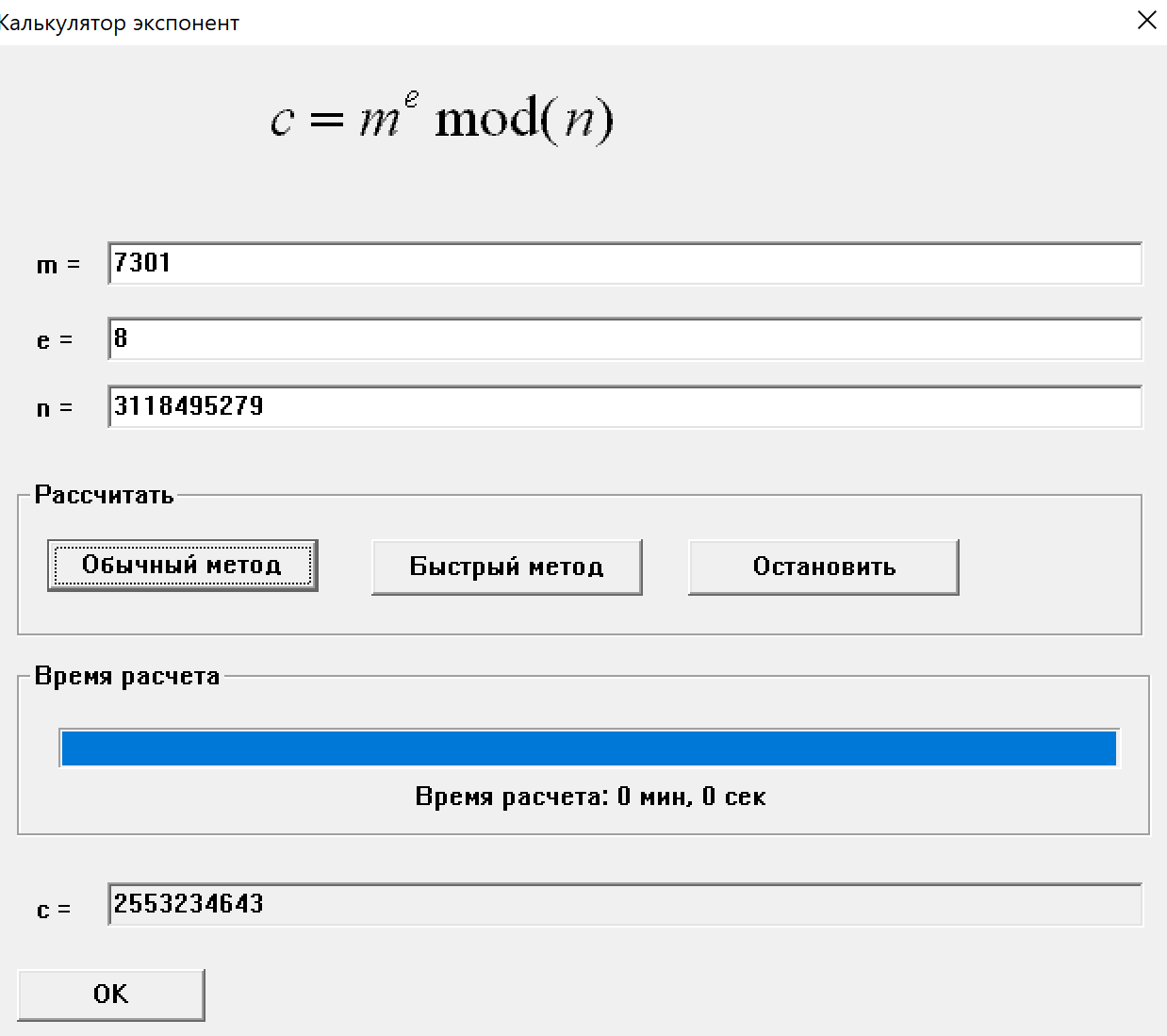
19. Получим подпись для хэш-образа , используя алгоритм модульного возведения в степень. Подписью будет пара (m,s). Верифицировать подпись проверкой сравнения .



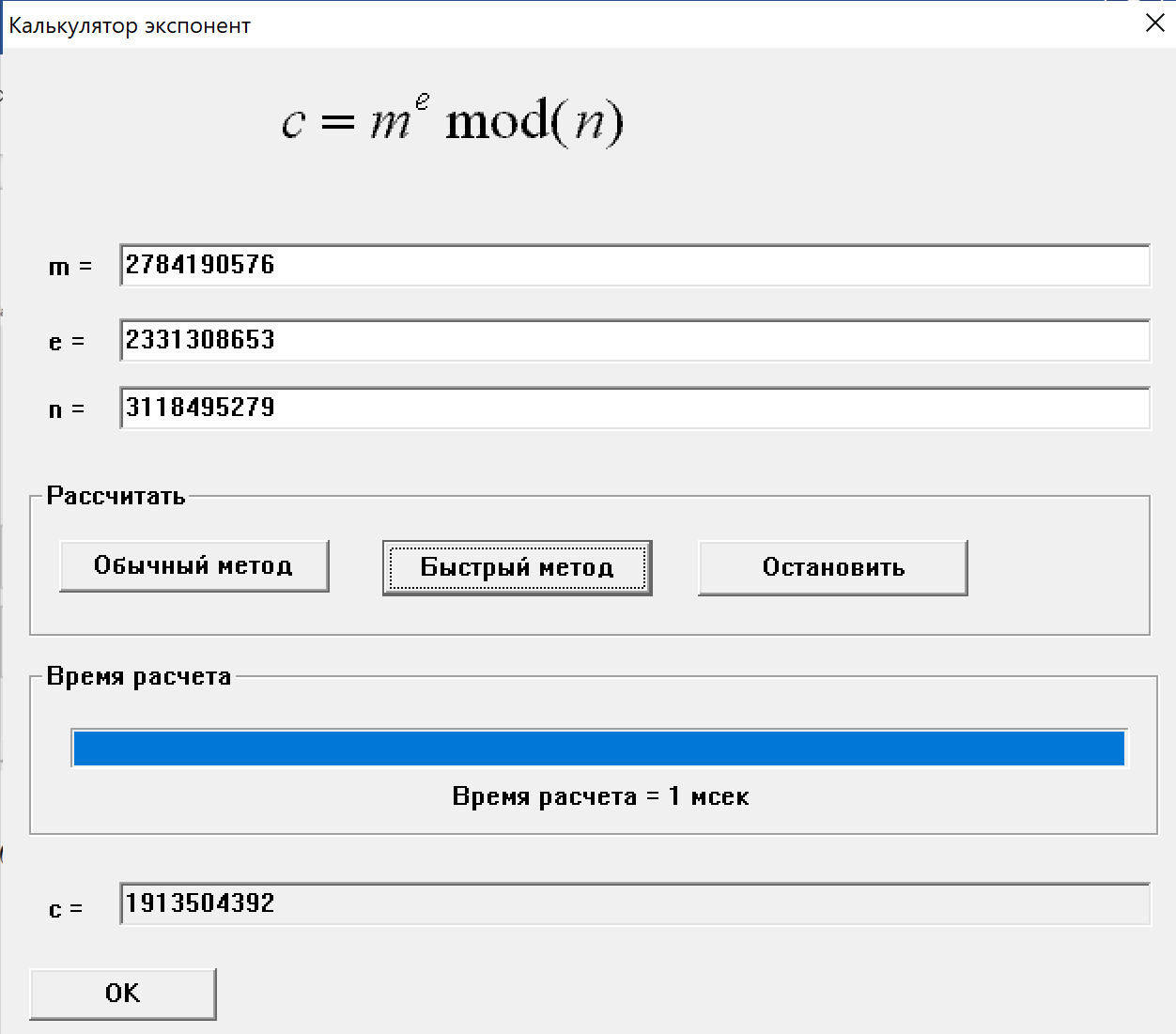


Подпись (7301, )

20. Создаем подделку цифровой подписи путем возведения *m* и *s* в степень с произвольным показателем *l*: *m’=mlmodn, s’=slmodn*. (*l* = 8).



21. Верифицируем подделанную подпись путем проверки сравнения *h(m’)* = *s’emodn.*



Использование хэширования обеспечивает безопасность цифровой подписи открытого текста и его целостности.

Вывод: в ходе работы была изучена криптосистема RSA и закреплены знания, полученные в ходе лекций, а также произведена проверка шифрования по алгоритму РША с хешированием и без него.