**Вариант 1**

Разработать программное средство, выполняющее вычисление открытого ключа **(*KO*)** алгоритма ***RSA*** и побайтовое шифрование данным ключом по алгоритму ***RSA*** произвольного файла. Значения параметров ***p,*** ***q*** и ***KС*,** а также имя входного файла задаются пользователем. Программа должна осуществлять проверку ограничений на вводимые пользователем значения параметров алгоритма. Организовать вывод содержимого зашифрованного файла на экран в виде чисел в 10 системе счисления.

Разработать программное средство, выполняющее расшифрование файла, каждый 16-битный блок которого представляет собой зашифрованное по алгоритму ***RSA*** 8-битное значение. Значения модуля ***r*** и ***закрытого*** ключа ***KС***задаются пользователем.

Использовать алгоритм быстрого возведения в степень и расширенный алгоритм Евклида.

При использовании длинной арифметики для определения простоты числа использовать один из вероятностных тестов: тест Ферма или тест Миллера-Рабина.

Результат работы программы – зашифрованный/расшифрованный файл/ы.

**Вариант 2**

Реализовать шифратор и дешифратор *алгоритма* ***Эль-Гамаля*** файла c произвольным содержимым, используя алгоритм быстрого возведения в степень, а также реализовать вычисление открытого ключа ***g*** при данном значении ***p***, используя алгоритм нахождения первообразного корня по модулю. Значения параметров ***p*, *x*** и ***k***задаются пользователем*.* Программа должна осуществлять проверку ограничений на вводимые пользователем значения параметров алгоритма. Организовать вывод содержимого зашифрованного файла на экран в виде чисел в 10-й системе счисления. Вывести значение ***g*** на экран. Результат работы программы – зашифрованный/расшифрованный файл/ы.

Используя алгоритм из методички, искать все первообразные корни по модулю ***p.*** Все найденные корни вывести на экран и предложить для шифрования ввести на выбор любой из найденных.

При использовании длинной арифметики для определения простоты числа использовать один из вероятностных тестов: тест Ферма или тест Миллера-Рабина.

**Вариант 3**

Реализовать шифратор и дешифратор по алгоритму ***Рабина*** (алгоритм из методички) файла с любым содержимым, используя расширенный алгоритм Евклида и алгоритм быстрого возведения в степень при дешифрации. Значения параметров ***p, q и (b)***задаются пользователем. Программа должна осуществлять проверку ограничений на вводимые пользователем значения параметров. Организовать вывод содержимого зашифрованного файла на экран в виде чисел в 10-й системе счисления. Результат работы программы – зашифрованный/расшифрованный файл/ы.

При использовании длинной арифметики для определения простоты числа использовать один из вероятностных тестов: тест Ферма или тест Миллера-Рабина.