Clase BigInteger

Usar el constructor problablePrime el cual dice que el numero elegido tiene una probabilidad de ser compuesto que no excede 2-100. Ó usar el constructor BigInteger expresando la certeza que queremos que consiga la cual excederá de 1 – 1/2certeza.(o lo que es lo mismo 1-2 -certeza).

Si pides que se genere una clave de manera automática, ¿quieres saber los primos p y q? Si es así, quieres que p y q tengan el mismo numero de bits? No,no? Cuantos bits quieres que tenga uno y que tenga otro? Maxima diferencia entre ambos según num de bits?

cuando se generan las claves automáticamente, la clave publica que numero de bits quieres que tenga? Quieres que se de la opción para 65537?

A nivel de teoría dada en Fundamentos de Seguridad, que importancia hay en las claves publicas parejas?

Se recomendaran primos seguros? O se ignoran en esta aplicación?

Como hacemos con los NNC, los calculamos siempre que se pidan? O por el contrario solo se calculan para números inferiores a X cantidad de bits? Si es asi, que bits o que tiempo ¿ Si siempre que lo pidan vamos a mostrarlos lo metemos en un fichero? Cuantos sacamos máximo por pantalla?

¿Para factorizar N que algoritmo uso para este proyecto? ¿Uso varios? De momento he usado Pollar rho factorizationg

Paradoja del cumpleaños

// comprobar, pero no sirve para nada ((w.multiply(s)).add(publica.multiply(t))).equals(BigInteger.ONE));

NUEVAS dudas

Preguntarle por el método calculateDistanceBits de la clase generarClaves

De cuantos bits hacer que se deje crear la clave automática? A partir de 6 bits ya aparecen claves distintas, pero mejor 8? Dejo que pueda ser publica e = privada d?