In [1]:
```
p=11; q=23;
```

In [4]:
```
n=p*q; n
```

Out[4]: 253

In [5]:
```
phi=(p-1)*(q-1); phi
```

Out[5]: 220

In [11]:
```
e=113 #escogemos un numero al azar entre 1 y 219
```

In [12]:
```
gcd(e,phi)
```

Out[12]: 1

In [13]:
```
d=inverse_mod(e,phi); d
```

Out[13]: 37

In [14]:
```
(n,e) #clave publica
```

Out[14]: (253, 113)

In [15]:
```
d #clave
```

Out[15]: 37

In [17]:
```
M=200 #Mensaje, un numero entre 0 y n-1, entre 0 y 252
```

In [19]:
```
C=M^e %n; C #ciframos. OJO: No lo hagais así
```

Out[19]: 140

In [20]:
```
C #numero cifrado que se envia
```

Out[20]: 140

In [21]:
```
M2=C^d %n; M2 #descriframos
```

Out[21]: 200

In [22]:
```
#por que digo no lo hagais asi???
```

In [24]:
```
764536978524697854376893542769875436987689245 3^78245378960245378542378025
4378904527890 % n
```

```
---------------------------------------------------------------------------
OverflowError                             Traceback (most recent call la
st)
<ipython-input-24-ecf76b0c0d58> in <module>
----> 1 Integer(764536978524697854376893542769875436987689245 3)**Integer
(78245378960245378542378025437890452789 0) % n

/opt/sagemath-9.2/local/lib/python3.7/site-packages/sage/rings/integer.p
yx in sage.rings.integer.Integer.__pow__ (build/cythonized/sage/rings/in
teger.c:15157)()
   2202
   2203         if type(left) is type(right):
-> 2204             return (<Integer>left)._pow_(right)
   2205         elif isinstance(left, Element):
   2206             return coercion_model.bin_op(left, right, operator.p
ow)

/opt/sagemath-9.2/local/lib/python3.7/site-packages/sage/rings/integer.p
yx in sage.rings.integer.Integer._pow_ (build/cythonized/sage/rings/inte
ger.c:15510)()
   2282                 r = smallInteger(1)
   2283         else:
-> 2284             raise OverflowError(f"exponent must be at most {LONG
_MAX}")
   2285         if mpz_sgn(exp) >= 0:
   2286             return r

OverflowError: exponent must be at most 9223372036854775807
```

In [25]:
```
power_mod(764536978524697854376893542769875436987689245 3,7824537896024537
8542378025437890452789 0,n)
```

Out[25]: 232

In [ ]:
```
#asi si se hace un exponente
```