RSA Encryption.

RSA encryption is a type of what is called “public key encryption”. Public key encryption works as follows. We start with some mathematical operation which is easy to do, but impossible to efficiently undo without some extra information. Then we make public the steps for encryption, but keep the extra information needed to decode messages secret. The result is that anyone can encrypt a message, but only we can decrypt messages.

In RSA encryption, the easy mathematical operation is multiplication, and the more difficult operation is factoring. Given two prime numbers, we can easily multiply them. On the other hand, given a very large (100+ digits) composite number, we have no better way of factoring it than simply testing every possible factor (a very slow algorithm). Here are the steps.

1. Choose large prime numbers p, q. Define $n = pq$.
2. Choose $e, d \in \mathbb{N}$ so that $ed \equiv 1 \mod (p-1)(q-1)$.
3. Make public the information n, e. Keep d, p, q secret.
4. To encode a message M, compute $C \equiv M^e \mod n$.
5. To decode a message C, compute $M \equiv C^d \mod n$.

Consider the following simple cipher.

A	B	C	D	E	F	G	H	I	L	M	N	O	P	R	S	T	U	W	Y
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

1. First, try out decryption. Suppose I take $p = 5$, $q = 7$, $e = 5$, $d = 5$. Decrypt the following message:
$$\underline{12}\ \underline{8}\ \underline{10}\ \underline{13}\ \underline{15}\ \underline{10}\ \underline{16}.$$
2. Now try encoding a short message (probably just a word), and trade with a group member. Have your group member decode your message.
3. **Why does this work?** Prove that $M^{ed} \equiv M \mod n$.
4. **(Challenge)** Choose 2 primes which are larger than 100, and your own cipher. Write a message for me, tell me n and e are, and what the cipher is, but do not tell me d, p, or q. I will try and break your code!