1) a) RTS: Given $n=pq$, $p>q$ and $\phi(n)=(p-1)(q-1)$, prove:

i)
$$p+q=n-\phi(n)+1$$
$$p+q=n-(p-1)(q-1)+1$$
$$p+q=n-(pq-p-q+1)+1$$
$$p+q=n-pq+p+q-1+1$$
$$p+q=n-n+p+q$$
$$p+q=p+q$$
❑

ii)
$$p-q=\sqrt{(p+q)^2-4n}$$
$$p-q=\sqrt{p^2+2pq+q^2-4pq}$$
$$p-q=\sqrt{p^2-2pq+q^2}$$
$$p-q=\sqrt{(p-q)^2}$$
$$p-q=p-q$$
❑

b) Factoring an integer into its prime factors is a *hard* problem (NP, actually) – kind of the idea behind encryption. With this being stated, I will instead write a program to factor it for me:

```
#include <iostream>
using namespace std;

int main() {
    int to_factor = 71531;
    for (int i = 3; i * i <= to_factor; i += 2)
        if (to_factor % i == 0) {
            cout << "Factors are " << i << " and "
                 << (to_factor / i) << endl;
            return 0;
        }
    return 1;
}
```

The output is “Factors are 233 and 307 ”. We can verify that by multiplying them together (successfully obtaining 71531), and multiplying their decrements (getting 70992). Since these are the numbers that were given, they must be the correct factors.

2) RTS: 2479 is not a prime number (without brute forcing).
BWOC, assume 2479 is a prime number. FLT states that any number coprime to 2479, raised to $2479-1 (mod\ 2479)$, will be 1 – if 2479 is a prime. We will square and multiply to find out:

$$2^2 \equiv 4 \pmod{2479}$$
$$2^4 \equiv 16 \pmod{2479}$$
$$2^8 \equiv 256 \pmod{2479}$$
$$2^{16} \equiv 1082 \pmod{2479}$$
$$2^{32} \equiv 636 \pmod{2479}$$
$$2^{64} \equiv 419 \pmod{2479}$$
$$2^{128} \equiv 2031 \pmod{2479}$$
$$2^{256} \equiv 2384 \pmod{2479}$$
$$2^{512} \equiv 1588 \pmod{2479}$$
$$2^{1024} \equiv 601 \pmod{2479}$$
$$2^{2048} \equiv 1746 \pmod{2479}$$
$$2^{2478} \equiv 1935 \pmod{2479}$$

Since this final mod is not 1, 2479 must not be prime. ❑

3) a) Want: 2762 encrypted by public key (21,12193):

$$2762^2 \equiv 8019 \pmod{12193}$$
$$2762^4 \equiv 10672 \pmod{12193}$$
$$2762^8 \equiv 8974 \pmod{12193}$$
$$2762^{16} \equiv 1426 \pmod{12193}$$
$$2762^{21} = 2762^{16} \cdot 2762^4 \cdot 2762$$
$$2762^{21} \equiv 11522 \pmod{12193}$$

b) Want 765 decrypted by private key (871, 1147). STSH: $765^{871} \bmod 1147$

$$765^2 \equiv 255 \pmod{1147}$$
$$765^4 \equiv 793 \pmod{1147}$$
$$765^8 \equiv 293 \pmod{1147}$$
$$765^{16} \equiv 791 \pmod{1147}$$
$$765^{32} \equiv 7 \pmod{1147}$$
$$765^{64} \equiv 49 \pmod{1147}$$
$$765^{128} \equiv 107 \pmod{1147}$$
$$765^{256} \equiv 1126 \pmod{1147}$$
$$765^{512} \equiv 441 \pmod{1147}$$
$$765^{871} = 765^{512} \cdot 765^{256} \cdot 265^{64} \cdot 265^{32} \cdot 265^4$$
$$765^{871} \equiv 164 \pmod{1147}$$