Q1. (10 points) Is $4^{1536} \equiv 9^{4824}$ mod 35

First note that 35 = 5 x 7, i.e. a product of two primes. We know that $x^4 \equiv 1$ mod 5 and $x^6 \equiv 1$ mod 7 by fermat's little theorem.
This implies that $(x^4)^6 \equiv 1$ mod 5 and $(x^6)^4 \equiv 1$ mod 7. Or in other words 5, divides $x^{24}-1$ and so does 7. Since 5 and 7 are both primes, and they divide the same number $x^{24}-1$, then it must be the case that 5 x 7 = 35 also divides $x^{24}-1$. In other words, we have
$x^{24}-1 \equiv 0$ mod 35 or $x^{24} \equiv 1$ mod 35

We can now apply this to check if the problem statement is true.
$4^{1536} = (4^{24})^{64} \equiv (1)^{64} \equiv 1$ mod 35
$9^{4824} = (9^{24})^{201} \equiv (1)^{201} \equiv 1$ mod 35
Thus, both the numbers have the same remainder mod 35, and therefore the statement is true.

Q2. (10 points) Solve $x^{86} \equiv 6$ mod 29

Since 29 is prime, we can again apply fermat's theorem to solve the problem. That is, $x^{28} \equiv 1$ mod 29. This implies that
$x^{86} \equiv (x^{28})^3 x^2 \equiv x^2$ mod 29
So, we have to solve the equation $x^2 \equiv 6$ mod 29. This implies that $x^2 = 29t + 6$, i.e., we are looking for an integer multiple of 29, t, such that we get a square when we add to 29t. This is true for t=2, since 29x2+6 = 58+6 = 64. This implies that x=8 is a solution to the above equation.

Q3. (10 points) Prove that $gcd(F_{n+1},F_n)=1$, for $n \geq 1$, where F_n is the n-th Fibonacci element.

From the gcd theorem (Euclid's rule on pg 20 in the book), we have gcd(x,y) = gcd(x-y,y).
Therefore, $gcd(F_{n+1}, F_n)= gcd(F_{n+1} - F_n, F_n)$
But $F_{n+1} - F_n = (F_n + F_{n-1}) - F_n = F_{n-1}$. We substitute this into the equation above to get
$gcd(F_{n+1}, F_n)= gcd(F_{n+1} - F_n, F_n) = gcd(F_n, F_{n-1})$
Repeating this multiple times we get
$gcd(F_{n+1}, F_n) = gcd(F_n, F_{n-1}) = gcd(F_{n-1}, F_{n-2}) = \ldots. = gcd(F_2, F_1)$
But, we know that $F_1=1$, $F_2=1$, which implies that $gcd(F_2, F_1) = 1$.
Therefore, $gcd(F_{n+1}, F_n) = 1$

Q4. (10 points) Assume that the cost to multiply a n-bit integer with a m-bit integer is O(nm). Given integers x and y with n-bits and m-bits, respectively, give an efficient algorithm to compute x^y. Show that the method is correct, and analyze its running time.

Approach 1: Iterative

```
iterative (x, y)
  if y = Ø : return 1
  product = x
  for (i = 1; i < y; i++):
    product = product * x
  return product
```

For the purposes of both the iterative and the recursive algorithms:

- x is n bits long
- y is m bits long
- the value of y is 2^{m-1}. In other words, the value of y is 1 followed by some number of zeroes (in binary): 1Ø, 1ØØ, 1ØØØ, etc... This allows us to avoid the 'y is odd' case in the recursive algorithm, and makes calculations easier.

Also, we know that the time to multiply an n-bit number by an m-bit number is $O(n \cdot m)$, and the number of bits in the product is $O(n+m)$.

Step #	value	time	resulting # of bits in product
1	n^2	n^2	$2n$
2	n^3	$2n^2$	$3n$
3	n^4	$3n^2$	$4n$
4	n^5	$4n^2$	$5n$
5	n^6	$5n^2$	$6n$
...	...	...	...

The total time is the sum of times of all steps. This algorithm will perform $y-1$ steps. Each individual step can be expressed as $i \times n^2$, where i is the step number. Therefore, the total time is:

$$\sum_{i=1}^{y-1} i \cdot n^2 = n^2 \cdot \sum_{i=1}^{y-1} = n^2 \cdot \frac{y(y-1)}{2} \approx n^2 y^2$$

$$y = 2^{m-1}$$

$$n^2 y^2 = n^2 \cdot \left(2^{m-1}\right)^2 = n^2 \cdot 2^{2m-2}$$

$$O(2^{2m} \cdot n^2)$$

Approach 2: Recursive

```
recursive (x, y)
  if y = Ø : return 1
  z = recursive (x, ⌊y/2⌋)
  if y is even:
      return z^2
  else:
      return z^2 · x
```

step #	value	time	resulting # of bits in product (z)
1	x^2	n^2	$2n$
2	x^4	$4n^2$	$4n$
3	x^8	$16n^2$	$8n$
4	x^{16}	$64n^2$	$16n$
5	x^{32}	$256n^2$	$32n$
...	...	...	...

The total running time will be the sum of the times of all steps. We know that this recursive algorithm will perform $\log_2 y$ steps. Each individual step can be expressed as $2^{2(i-1)} \cdot n^2$, where i is the step number. Thus, the total running time can be expressed as:

$$\sum_{i=1}^{\log_2 y} 2^{2(i-1)} \cdot n^2$$

$$y = 2^{m-1}$$

$$\sum_{i=1}^{\log_2 y} = \sum_{i=1}^{\log_2 2^{m-1}} = \sum_{i=1}^{m-1}$$

$$\sum_{i=1}^{m-1} 2^{2(i-1)} \cdot n^2 = n^2 \cdot \sum_{i=1}^{m-1} 2^{2(i-1)} = n^2 \cdot (2^0 + 2^2 + 2^4 + 2^6 \dots + 2^{2m-4})$$

$$= n^2 \cdot (2^0 + 2^2 + 2^4 + 2^6 \dots + 2^{2m-4}) \cdot \left(\frac{2^2-1}{2^2-1}\right) =$$

$$= n^2 \cdot \left[\frac{(2^2 + 2^4 + 2^6 + 2^8 \dots + 2^{2m-4} + 2^{2m-2}) - (2^0 + 2^0 + 2^4 + 2^6 \dots + 2^{2m-4})}{2^2 - 1}\right]$$

$$= n^2 \cdot \frac{2^{2m-2} - 1}{2^2 - 1} = n^2 \cdot \frac{2^{2m-2} - 1}{3}$$

$$O(2^{2m} \cdot n^2)$$