CS 70 Discrete Mathematics and Probability Theory
Summer 2019 James Hulett and Elizabeth Yang

HW 3

1 Squared RSA

(a) (10 points) Prove the identity $a^{p(p-1)} \equiv 1 \pmod{p^2}$, where a is coprime to p, and p is prime. (Hint: Try to mimic the proof of Fermat's Little Theorem from the notes.)

(b) (10 points) Now consider the RSA scheme: the public key is $(N = p^2q^2, e)$ for primes p and q, with e relatively prime to $p(p-1)q(q-1)$. The private key is $d = e^{-1} \pmod{p(p-1)q(q-1)}$. Prove that the scheme is correct for x relatively prime to both p and q, i.e. $x^{ed} \equiv x \pmod{N}$.

Solution:

(a) We mimic the proof of Fermat's Little Theorem from the notes.
Let S be the set of all numbers between 1 and $p^2 - 1$ (inclusive) which are relatively prime to p. We can write

$$S = \{1, 2, \ldots, p-1, p+1, \ldots, p^2 - 1\}$$

Define the set

$$T = \{a, 2a, \ldots, (p-1)a, (p+1)a, \ldots, (p^2-1)a\}$$

We'll show that $S \subseteq T$ and $T \subseteq S$, allowing us to conclude $S = T$:

- $S \subseteq T$: Let $x \in S$. Since $\gcd(a, p) = 1$, the inverse of a exists $\pmod{p^2}$. For ease of notation, we use a^{-1} to denote the quantity $a^{-1} \pmod{p^2}$. We know $\gcd(a^{-1}, p) = 1$, because a^{-1} has an inverse $\pmod{p^2}$ too. Combining this with the fact that $\gcd(x, p) = 1$, we have $\gcd(a^{-1}x, p) = 1$. This tells us $a^{-1}x \in S$, so $a(a^{-1}x) = x \in T$.
- $T \subseteq S$: Let $ax \in T$, where $x \in S$. We know $\gcd(x, p) = 1$ because $x \in S$. Since $\gcd(a, p) = 1$ as well, we know the product xs cannot share any prime factors with p as well, i.e. $\gcd(xs, p) = 1$. This means $xs \in S$ as well, which proves the containment.

We now follow the proof of Fermat's Little Theorem. Since $S = T$, we have:

$$\prod_{s_i \in S} s_i \equiv \prod_{t_i \in T} t_i \pmod{p^2}$$

However, since we defined $T = \{a, 2a, \ldots, (p-1)a, (p+1)a, \ldots, (p^2-1)a\}$:

$$\prod_{t_i \in T} t_i \equiv \prod_{s_i \in S} a s_i \equiv a^{|S|} \prod_{s_i \in S} s_i \pmod{p^2}$$

We can now conclude $(\prod_{s_i \in S} s_i) \equiv a^{|S|}(\prod_{s_i \in S} s_i) \pmod{p^2}$.
Each $s_i \in S$ is coprime to p, so their product $\prod_{s_i \in S} s_i$ is as well. Then, we can multiply both sides of our equivalence with the inverse of $\prod_{s_i \in S} s_i$ to obtain $a^{|S|} \equiv 1 \pmod{p^2}$. Using HW4, 4(b), we know $|S| = p(p-1)$, which gives the desired result.

Alternate Solution: We can use Fermat's Little Theorem, combined with the Binomial Theorem, to get the result. Since $\gcd(a,p) = 1$ and p is prime, $a^{p-1} \equiv 1 \pmod{p}$, so we can write $a^{p-1} = \ell p + 1$ for some integer ℓ. Then,

$$(a^{p-1})^p = (\ell p + 1)^p = \sum_{i=0}^{p} \binom{n}{i} (\ell p)^i = 1 + p \cdot (\ell p) + \binom{p}{2} (\ell p)^2 + \cdots + (\ell p)^p,$$

and since all of the terms other than the first term are divisible by p^2, $a^{p(p-1)} \equiv 1 \pmod{p^2}$.

(b) By the definition of d above, $ed = 1 + kp(p-1)q(q-1)$ for some k. Look at the equation $x^{ed} \equiv x \pmod{N}$ modulo p^2 first:

$$x^{ed} \equiv x^{1+kp(p-1)q(q-1)} \equiv x \cdot (x^{p(p-1)})^{kq(q-1)} \equiv x \pmod{p^2}$$

where we used the identity above. If we look at the equation modulo q^2, we obtain the same result. Hence, $x^{ed} \equiv x \pmod{p^2 q^2}$.

Remark: The first part of the question mirrors the proof of Fermat's Little Theorem. The second and third parts of the question mirror the proof of correctness of RSA.

2 Breaking RSA (15 points)

Eve is not convinced she needs to factor $N = pq$ in order to break RSA. She argues: "All I need to know is $(p-1)(q-1)$... then I can find d as the inverse of e mod $(p-1)(q-1)$. This should be easier than factoring N." Prove Eve wrong, by showing that if she knows $(p-1)(q-1)$, she can easily factor N (thus showing finding $(p-1)(q-1)$ is at least as hard as factoring N). Assume Eve has a friend Wolfram, who can easily return the roots of polynomials over $\mathbb{R}$ (this is, in fact, easy).

Solution:

Let $a = (p-1)(q-1)$. If Eve knows $a = (p-1)(q-1) = pq - (p+q) + 1$, then she knows $p + q = pq - a + 1$ (note that $pq = N$ is known too). In fact, p and q are the two roots of polynomial $f(x) = x^2 - (p+q)x + pq$ because $x^2 - (p+q)x + pq = (x-p)(x-q)$. Since she knows $p+q$ and pq, she can give the polynomial $f(x)$ to Wolfram to find the two roots of $f(x)$, which are exactly p and q.

Alternate Solution: Consider the polynomial $r(x) = (x-p)(x-q)$. Evaluate the polynomial at three special points.

$$r(0) = N$$
$$r(1) = (p-1)(q-1)$$
$$r(N) = N(p-1)(q-1)$$

Use polynomial interpolation to find the polynomial that goes through the three points $(0, N)$, $(1, (p-1)(q-1))$, $(N, N(p-1)(q-1))$, and then ask Wolfram for the roots of the polynomial.

3 Lagrange? More like Lamegrange.

In this problem, we walk you through an alternative to Lagrange interpolation.

(a) (5 points) Let's say we wanted to interpolate a polynomial through a single point, (x_0, y_0). What would be the polynomial that we would get? (This is not a trick question.)

(b) (5 points) Call the polynomial from the previous part $f_0(x)$. Now say we wanted to define the polynomial $f_1(x)$ that passes through the points (x_0, y_0) and (x_1, y_1). If we write $f_1(x) = f_0(x) + a_1(x - x_0)$, what value of a_1 causes $f_1(x)$ to pass through the desired points?

(c) (5 points) Now say we want a polynomial $f_2(x)$ that passes through (x_0, y_0), (x_1, y_1), and (x_2, y_2). If we write $f_2(x) = f_1(x) + a_2(x - x_0)(x - x_1)$, what value of a_2 gives us the desired polynomial?

(d) (5 points) Suppose we have a polynomial $f_i(x)$ that passes through the points (x_0, y_0), ..., (x_i, y_i) and we want to find a polynomial $f_{i+1}(x)$ that passes through all those points and also (x_{i+1}, y_{i+1}). If we define $f_{i+1}(x) = f_i(x) + a_{i+1} \prod_{j=0}^{i} (x - x_j)$, what value must a_{i+1} take on?

Solution:

(a) We want a degree zero polynomial, which is just a constant function. The only constant function that passes through (x_0, y_0) is $f_0(x) = y_0$.

(b) By defining $f_1(x) = f_0(x) + a_1(x - x_0)$, we get that

$$f_1(x_0) = f_0(x_0) + a_1(x_0 - x_0) = y_0 + 0 = y_0.$$

So now we just need to make sure that $f_1(x_1) = y_1$. This means that we need to choose a_1 such that

$$f_1(x_1) = f_0(x_1) + a_1(x_1 - x_0) = y_1.$$

Solving this for a_1, we get that

$$a_1 = \frac{y_1 - f_0(x_1)}{x_1 - x_0}.$$

(c) We apply similar logic to the previous part. From our definition, we know that

$$f_2(x_0) = f_1(x_0) + a_2(x_0 - x_0)(x_0 - x_1) = y_0 + 0 = y_0.$$

and that

$$f_2(x_1) = f_1(x_1) + a_2(x_1 - x_0)(x_1 - x_1) = y_1 + 0 = y_1.$$

Thus, we just need to choose a_2 such that $f_2(x_2) = y_2$. Putting in our formula for $f_2(x)$, we get that we need a_2 such that

$$f_1(x_2) + a_2(x_2 - x_0)(x_2 - x_1) = y_2.$$

Solving for a_2, we get that

$$a_2 = \frac{y_2 - f_1(x_2)}{(x_2 - x_0)(x_2 - x_1)}.$$

(d) If we try to calculate $f_{i+1}(x_k)$ for $0 \leq k \leq i$, we know one of the $(x - x_j)$ terms (specifically the kth one) will be zero. Thus, we get that

$$f_{i+1}(x_k) = f_i(x_k) + a_{i+1}(0) = y_k + 0 = y_k.$$

So now we just need to pick a_i such that $f_{i+1}(x_{i+1}) = y_{i+1}$. This means that we need to choose a_{i+1} such that

$$f_i(x_{i+1}) + a_{i+1} \prod_{j=0}^{i} (x_{i+1} - x_j) = y_{i+1}.$$

Solving for a_{i+1}, we get that

$$a_{i+1} = \frac{y_{i+1} - f_i(x_{i+1})}{\prod_{j=0}^{i}(x_{i+1} - x_j)}.$$

The method you derived in this question is known as Newtonian interpolation. (The formal definition of Newtonian interpolation uses divided differences, which we don't cover in this class, but it's in effect doing the same thing.) This method has an advantage over Lagrange interpolation in that it is very easy to add in extra points that your polynomial has to go through (as we showed in part (c), whereas Lagrange interpolation would require you to throw out all your previous work and restart. However, if you want to keep the same x values but change the y values, Newtonian interpolation requires you to throw out all your previous work and restart. In contrast, this is fairly easy to do with Lagrange interpolation–since changing the y values doesn't affect the δ_is, you don't have to recalculate those, so you can skip most of the work.

4 Error-Correcting Polynomials

(a) (5 points) Alice has a length 8 message to Bob. There are 2 communication channels available. When n packets are fed through channel A, the channel will only deliver 5 packets (picked at random). Similarly, channel B will only deliver 5 packets (picked at random), but it will also corrupt (change the value) of one of the delivered packets. Each channel will only work if at least 10 packets are sent through it. Using each of the 2 channels once, how can Alice send the message to Bob?

(b) (5 points) Alice wishes to send a message to Bob as the coefficients of a degree 2 polynomial P. For a message $[m_1, m_2, m_3]$, she creates polynomial $P = m_1 x^2 + m_2 x + m_3$ and sends 5 packets: $(0, P(0)), (1, P(1)), (2, P(2)), (3, P(3)), (4, P(4))$. However, Eve interferes and changes one of the values of a packet before it reaches Bob. If Bob receives

$$(0,3), (1,0), (2,3), (3,0), (4,3),$$

and knows Alice's encoding scheme and that Eve changed one of the packets, can he still figure out what the original message was? If so find it as well as the x-value of the packet that Eve changed, if not, explain why he can not. (Work in mod 11.)

(c) (5 points) Alice decides that putting the message as the coefficients of a polynomial is too inefficient for long messages because the degree of the polynomial grows quite large. Instead, she decides to encode the message as values in a degree 2 polynomial. For a 5 length message $[m_0, m_1, m_2, m_3, m_4]$, she creates a degree 2 polynomial P such that $P(0) = m_0, P(1) = m_1, P(2) = m_2, P(3) = m_3, P(4) = m_4$. (Alice makes sure to choose her message in such a way that it can be encoded in a polynomial of degree 2.) She then sends the length 5 message directly to Bob as 5 packets: $(0, m_0), (1, m_1), (2, m_2), (3, m_3), (4, m_4)$. Eve again interfere and changes the value of a packet before it reaches Bob. If Bob receives $(0,0), (1,3), (2,0), (3,3), (4,0)$ and knows Alice's encoding scheme and that Eve changed one of the packets, can he still figure out what the original message was? If so find it as well as the x-value of the packet that Eve changed, if not, explain why he can not. (Work in mod 11.)

(d) (10 points) After getting tired of decoding degree 2 polynomials, Bob convinces Alice to send messages using a degree 1 polynomial instead. To be on the safer side, Alice decides to continue to send 5 points on the polynomial even though it is only degree 1. She encodes and sends a length 5 message in the same way as part (c) (except using a degree 1 polynomial). Eve however, decides to change 2 of the packets. After Eve interferes, Bob receives $(0,-3), (1,-1), (2,x), (3,-3), (4,5)$. If Alice sent $(0,-3), (1,-1), (2,1), (3,3),$ $(4,5)$, for what values of x will Bob not be able to uniquely determine the Alice's message? (Assume Bob knows that Eve changed 2 of the packets and **work in mod 13.**)

Solution:

(a) Channel A will deliver 5 packets so we can send a message of length 5 encoded on a polynomial of degree 4 though it. If we send 10 points though channel A, it doesn't matter which 5 points bob gets, he will still be able to reconstruct our degree 4 polynomial. Since the channel B has 1 general error, we can only send a message of length 3 encoded on a degree 2 polynomial through it. If we send 10 points, Bob will get 5 points to calculate a degree 2 polynomial with 1 general error, which he is able to do. Thus to send our length 8 message, we can send the character 1 - 5 through a channel A and the characters 6 - 8 through channel B.

(b) We can use Berlekamp and Welch. We have: $Q(x) = P(x)E(x)$. $E(x)$ has degree 1 since we know we have at most 1 error. $Q(x)$ is degree 3 since $P(x)$ is degree 2. We can write a system of linear equations and solve:

$$\begin{aligned}
d &= 3(0-e) \\
a+b+c+d &= 0(1-e) \\
8a+4b+2c+d &= 3(2-e) \\
27a+9b+3c+d &= 0(3-e) \\
64a+16b+4c+d &= 3(4-e)
\end{aligned}$$

Since we are working in mod 11, this is equivalent to:

$$\begin{aligned}
d &= -3e \\
a+b+c+d &= 0 \\
8a+4b+2c+d &= 6-3e \\
5a+9b+3c+d &= 0 \\
9a+5b+4c+d &= 1-3e
\end{aligned}$$

Solving yields:

$$Q(x) = x^3+5x^2+5, E(x) = x-2$$

To find $P(x)$ we divide $Q(x)$ by $E(x)$ and get $P(x) = x^2+7x+3$. So Alice's message is $m_1 = 1, m_2 = 7, m_3 = 3$.

Alternative solution: Since we have 5 points, we have to find a polynomial of degree 2 that goes through 4 of those points. The point that the polynomial does not go through will be the packet that Eve changed. Since 3 points uniquely determine a polynomial of degree 2, we can pick 3 points and check if a 4th point goes through it. (It may be the case that we need to try all sets of 3 points.) We pick the points $(0,3),(1,0),(3,0)$. Lagrange interpolation can be used to create the polynomial but we can see that for the polynomial that goes through these 3 points, it has 0's at $x = 1$ and $x = 3$. Thus the polynomial is $(x-1)(x-3) = x^2-4x+3 \pmod{11} = x^2+7x+3 \pmod{11}$. We then check to see if the this polynomial goes through one of the 2 points that we didn't use. Plugging in 4 for x, we get 3. The packet that Eve changed is the point that our polynomial does not go through which has x-value 2. Alice's original message was $m_1 = 1, m_2 = 7, m_3 = 3$.

(c) To find the polynomial P, we can use Berlekamp and Welch. We have: $Q(x) = P(x)E(x)$. $E(x)$ has degree 1 since we know we have at most 1 error. $Q(x)$ is degree 3 since $P(x)$ is degree 2. We can write a system of linear equations and solve:

$$d = 0(0-e)$$

$$a+b+c+d = 3(1-e)$$

$$8a+4b+2c+d = 0(2-e)$$

$$27a+9b+3c+d = 3(3-e)$$

$$64a+16b+4c+d = 0(4-e)$$

Since we are working in mod 11, this is equivalent to:

$$d = 0$$

$$a+b+c+d = 3-3e$$

$$8a+4b+2c+d = 0$$

$$5a+9b+3c+d = 9-3e$$

$$9a+5b+4c+d = 0$$

Solving yields:

$$Q(x) = -x^3+6x^2-8x \pmod{11} = 10x^3+6x^2+3x \pmod{11}, E(x) = x-2$$

To find $P(x)$ we divide $Q(x)$ by $E(x)$ and get $P(x) = 10x^2+4x$. To recover Alice's message, we evaluate the polynomial at: $0,1,2,3,4$ and get the message: $0,3,4,3,0$. The x-value of the packet that Eve changed is 2 (given by $E(x)$).

(d) Since Bob knows that Eve changed 2 of the points, the 3 remaining points will still be on the degree 1 polynomial that Alice encoded her message on. Thus if Bob can find a degree 1 polynomial that passes through at least 3 of the points that he receives, he will be able to uniquely recover Eve's message. The only time that Bob cannot uniquely determine Alice's message is if there are 2 polynomials with degree 1 that passes through 3 of the 5 points that he receives. Since we are working with degree 1 polynomials, we can plot the points that Bob receives and then see which values of x will cause 2 sets of 3 points to fall on a line. $(0,-3),(1,-1),(4,5)$ already fall on a line. If $x=-2$, $(1,-1),(2,-2),(3,-3)$ also falls on a line. If $x=-3$, $(0,-3),(2,-3),(3,-3)$ also falls on a line. If $x=1$, $(0,-3),(2,1),(4,5)$ falls on the original line, so here Bob can decode the message. If $x=2$, $(2,2),(3,-3),(4,5)$ also falls on a line. So if $x=-3,-2,2$, Bob will not be able to uniquely determine Alice's message.

5 Berlekamp-Welch Algorithm

In this question we will send the message $(m_0,m_1,m_2) = (4,3,2)$ of length $n=3$. We will use an error-correcting code for $k=1$ general error, doing arithmetic over GF(5).

(a) (5 points) Construct a polynomial $P(x) \pmod 5$ of degree at most 2, so that

$$P(0) = 4, \qquad P(1) = 3, \qquad P(2) = 2.$$

What is the message (c_0,c_1,c_2,c_3,c_4) that is sent?

(b) (5 points) Suppose the message is corrupted by changing c_0 to 0. Set up the system of linear equations in the Berlekamp-Welch algorithm to find $Q(x)$ and $E(x)$.

(c) (5 points) Assume that after solving the equations in part (b) we get $Q(x) = -x^2 + 4x$ and $E(x) = x$. Show how to recover the original message from Q and E.

Solution:

(a) We use Lagrange interpolation to construct the unique quadratic polynomial $P(x)$ such that $P(0) = m_0 = 4, P(1) = m_1 = 3, P(2) = m_2 = 2$.

$$\begin{aligned}
\Delta_0(x) &= \frac{(x-1)(x-2)}{(0-1)(0-2)} = \frac{x^2-3x+2}{2} \\
\Delta_1(x) &= \frac{(x-0)(x-2)}{(1-0)(1-2)} = \frac{x^2-2x}{-1} \\
\Delta_2(x) &= \frac{(x-0)(x-1)}{(2-0)(2-1)} = \frac{x^2-x}{2} \\
P(x) &= m_0\Delta_0(x) + m_1\Delta_1(x) + m_2\Delta_2(x) \\
&= 4\Delta_0(x) + 3\Delta_1(x) + 2\Delta_2(x) \\
&= -x+4
\end{aligned}$$

[Note that all arithmetic is over GF(5), so for example $2^{-1} \equiv 3 \pmod 5$.] For the final message we need to add 2 redundant points of P. Since 3 and 4 are the only points in GF(5) that we have not used yet, we compute $P(3) = 1, P(4) = 0$, and so our message is $(4,3,2,1,0)$.

(b) The message received is $(c_0', c_1', c_2', c_3', c_4') = (0,3,2,1,0)$. Let $R(x)$ be the function such $R(i) = c_i'$ for $0 \le i < 5$. Let $E(x) = x + b_0$ be the error-locator polynomial, and $Q(x) = P(x)E(x) = a_3x^3 + a_2x^2 + a_1x + a_0$. Since $Q(i) = P(i)E(i) = R(i)E(i)$ for $1 \le i < 5$, we have the following equalities $\pmod 5$:

$$\begin{aligned}
Q(0) &= 0E(0) \\
Q(1) &= 3E(1) \\
Q(2) &= 2E(2) \\
Q(3) &= 1E(3) \\
Q(4) &= 0E(4)
\end{aligned}$$

They lead to the following system of linear equations:

$$\begin{array}{rcrcrcrcrcl}
 & & & & & & a_0 & & & = & 0 \\
a_3 & + & a_2 & + & a_1 & + & a_0 & - & 3b_0 & = & 3 \\
8a_3 & + & 4a_2 & + & 2a_1 & + & a_0 & - & 2b_0 & = & 4 \\
27a_3 & + & 9a_2 & + & 3a_1 & + & a_0 & - & b_0 & = & 3 \\
64a_3 & + & 16a_2 & + & 4a_1 & + & a_0 & & & = & 0
\end{array}$$

(c) From the solution, we know

$$\begin{aligned}
Q(x) &= a_3x^3 + a_2x^2 + a_1x + a_0 = -x^2 + 4x, \\
E(x) &= x + b_0 = x.
\end{aligned}$$

Since $Q(x) = P(x)E(x)$, the recipient can compute $P(x) = Q(x)/E(x) = -x+4$ [note that this is the same polynomial $P(x)$ from part (a) used by the sender]. The recipient may deduce the location of the error from $E(x)$ as follows. There is only one error at location e_1, we have $E(x) = (x-e_1) = x$, so $e_1 = 0$ and the error is at position 0. To correct the error we evaluate $P(0) = 4$. Since the other two positions m_1, m_2 of the message are uncorrupted, we recover the original message $(m_0, m_1, m_2) = (4, 3, 2)$.

6 Countability Practice

(a) (10 points) Prove or disprove: The set of increasing functions $f : \mathbb{N} \to \mathbb{N}$ (i.e., if $x \geq y$, then $f(x) \geq f(y)$) is countable.

(b) (10 points) Prove or disprove: The set of decreasing functions $f : \mathbb{N} \to \mathbb{N}$ (i.e., if $x \geq y$, then $f(x) \leq f(y)$) is countable.

(c) (5 points) Is a set of disks in $\mathbb{R}^2$ such that no two disks overlap necessarily countable or possibly uncountable? [A disk is a region in the plane of the form $\{(x,y) \in \mathbb{R}^2 : (x-x_0)^2 + (y-y_0)^2 \leq r^2\}$, for some $x_0, y_0, r \in \mathbb{R}$, $r > 0$.]
(Hint: Try to relate it to something we know that's countable, such as $\mathbb{Q} \times \mathbb{Q}$)

(d) (5 points) Is a set of circles in $\mathbb{R}^2$ such that no two circles overlap necessarily countable or possibly uncountable? [*Hint*: A circle is a subset of the plane of the form $\{(x,y) \in \mathbb{R}^2 : (x-x_0)^2 + (y-y_0)^2 = r^2\}$ for some $x_0, y_0, r \in \mathbb{R}$, $r > 0$. The difference between a circle and a disk is that a disk contains all of the points in its interior, whereas a circle does not.]

Solution:

(a) Suppose that there is a bijection between $\mathbb{N}$ and the set of all increasing functions $\mathbb{N} \to \mathbb{N}$:

$$\begin{aligned} 0 &\mapsto (f_0(0), f_0(1), f_0(2), \ldots) \\ 1 &\mapsto (f_1(0), f_1(1), f_1(2), \ldots) \\ 2 &\mapsto (f_2(0), f_2(1), f_2(2), \ldots) \\ &\vdots \end{aligned}$$

We will use a diagonalization argument to prove that there is a function f which is not in the above list. Define

$$f(n) = 1 + \sum_{i=1}^{n} f_i(n).$$

First, we will show that f is increasing. Indeed, if $m \leq n$, then

$$f(m) = 1 + \sum_{i=1}^{m} f_i(m) \leq 1 + \sum_{i=1}^{n} f_i(m) \leq 1 + \sum_{i=1}^{n} f_i(n) = f(n).$$

The first inequality is because each function is non-negative; the second inequality is because the f_i are increasing.

To show that f is not in the list, note that

$$f(n) = 1 + \sum_{i=1}^{n} f_i(n) \geq 1 + f_n(n) > f_n(n).$$

Since $f(n) > f_n(n)$ for each $n \in \mathbb{N}$, f cannot be any of the functions in the list. Therefore, the set of increasing functions $f : \mathbb{N} \to \mathbb{N}$ is uncountable.

(b) Given any function that begins with $f(0) = n$, consider the number of indices in which the function decreases in output: the set of i such that $f(i) < f(i-1)$. The range of f is a subset of $\mathbb{N}$ so by the well-ordering principle there must be a least element. Call this element a. Then there are only at most $n-a$ transition points. We can set a bijection for any function with $f(0) = n$ to a "word" of indices at which the function decreases. Therefore, the set of decreasing functions $\mathbb{N} \to \mathbb{N}$ has the same cardinality as the set of finite bit strings from a countably infinite alphabet, which is countable. Therefore, the set of all decreasing functions is countable.

(c) Countable. Each disk must contain at least one rational point (an (x,y)-coordinate where $x, y \in \mathbb{Q}$) in its interior, and due to the fact that no two disks overlap, the cardinality of the set of disks can be no larger than the cardinality of $\mathbb{Q} \times \mathbb{Q}$, which we know to be countable.

(d) Possibly uncountable. Consider the circles $C_r = \{(x,y) \in \mathbb{R}^2 : x^2 + y^2 = r\}$ for each $r \in \mathbb{R}$. For $r_1 \neq r_2$, C_{r_1} and C_{r_2} do not overlap, and there are uncountably many of these circles (one for each real number).