Q1. Two-time pad

1. We know that $C_1 = P_1 \oplus K$ and $C_2 = P_2 \oplus K$.
   When we XOR C_1 and C_2, we get:
   $$C_1 \oplus C_2 = (P_1 \oplus K) \oplus (P_2 \oplus K)$$
   $$C_1 \oplus C_2 = P_1 \oplus P_2 \oplus K \oplus K$$
   $C_1 \oplus C_2 = P_1 \oplus P_2$, since $K \oplus K$ = 0, and $X \oplus 0 = X$ for any bitstring X.

2. For each byte of $C_1 \oplus C_2$, construct a list of all possible pairs of P_1 and P_2. We can brute-force our way through to create a combination such that P_1 and P_2 outputs some meaningful text. In addition to that, we can use the most frequently appearing characters that appear in English words, and try those characters first, to improve search performance.

Q2. Relay Selection in Anonymity Networks

(a) If the attacker uses the guard relay, then the user's IP address will be exposed to the attacker. If the attacker uses the exit relay, then the original message can be read and modified. Also, the attacker can reply to it with some malicious message.

(b) First, it provides anonymity unlinkably. Receiving multiple Tor connections within long periods of time (eg. 1 day difference) will make the source of them (user) undistinguishable, thus enhancing protection to user's privacy.
Second, it provides anonymity linkably. Opposing to the previous example, receiving multiple Tor connections within short periods of time (eg. 5 minute difference) will support the notion that the source of them is the same, thus weakening protection to user's privacy.

(c) If a user consistently uses the same email account over the anonymity network, then such successful connections will generate a recurring pattern, thus allows the attacker to track the user's connection information. This issue persists even with periodic changes on the user's privacy.

(d) Advantage: User experience stable, consistent performance and behaviour of anonymity system.
Disadvantage: Attacker can analyze traffic patterns based on behaviour and connection timestamps within onion routers. Thus, the user is more exposed to security threats. As the attacker identifies a circuit, security is no longer ensured.

(e) Tor can provide stable user connection by maintaining such set of relays because it will reduce the risk of malicious entities to maintain them.

(f) Monitor packages that are exiting the guard node. These will have some exiting timing pattern. We are given that Tor relays do not delay or re-order packages, so the exiting pattern with be equivalent to the entering pattern (to/from the guard node). This property allows easier analysis of package patterns, and reason about the relationship between the source and destination.

(g) For circuit $n_1, n_2, \ldots, n_n$, the following factors allow the image to be published on the Tor network.
- Guard node:
  - Knows: Dog owner is using Tor.
  - Doesn't know: Website the dog owner is visiting.
- Exit node:
  - Knows: User X is using Tor to publish something.
  - Doesn't know: User X's identity.
- Publishing website:
  - Knows: Established connection from Tor circuit node n_n.
    - Note, this connection is not encrypted, and is the only component that is not encrypted, according to Tor's design. Can be encrypted using HTTPS, etc.

Thus, the information of the image publisher (eg. IP address) in the process of encrypting the request/replies and the image itself on all connections. We therefore have established anonymous publication.

Q3. Data Privacy

(a) Tracker attack.

Our objective is to find out Charlie's grade, which is by using the following query:

```
C =                SELECT SUM(Grade) FROM Student
                   WHERE Name = 'Charlie'
```

, which is unachievable if only used itself, because it doesn't satisfy the "k ≤ *num_returned* ≤ N-k" constraint. Thus, we design a tracker and set of 3 queries to derive Charlie's grades.

1. Design a tracker.

   ```
   T = SELECT SUM(Grade) FROM Student WHERE Gender = 'M'
   ```

2. 3 queries for tracker.

   ```
   C or T = SELECT SUM(Grade) FROM Student
                WHERE Name = 'Charlie' OR Gender = 'M'

   C or not T = SELECT SUM(Grade) FROM Student
                 WHERE Name = 'Charlie' OR Gender != 'M'

   A = SELECT SUM(Grade) FROM Student
   ```

3. Deriving Charlie's grade using queries.

   Notice that using the 3 queries, we can derive Charlie's grades:

   ```
   q(C) = q(C or T) + q(C or not T) - q(A)
   ```

   Explanation:

   - `q(C or T)`: Returns sum of grade entries of { Males }
   - `q(C or not T)`: Returns sum of grade entries of { Charlie, Females }
   - `q(A)`: Returns all entries
   - Thus,
     - `q(C or T) + q(C or not T)` = { Charlie, Males+Females=All }
     - `q(A)` = { All }
     - Subtracting `q(S)` from `q(C or T) + q(C or not T)` will give us sum of grade entries of { Charlie }, as desired.

(b) Execute a Binary Search, still using the tracker attack.

For the tracker `T1`:

```
T1 = SELECT COUNT(*) FROM Student
       WHERE Name = 'Natalie' and Score = mid
```

, where `mid=(low+high)/2`.

Set up 3 queries:

```
CorT1 = SELECT COUNT(*) FROM Student
         WHERE (Name = 'Natalie' AND Score = mid) OR Gender = 'M'
CorNotT1 = SELECT COUNT(*) FROM Student
         WHERE (Name = 'Natalie' AND Score = mid) OR Gender != 'M'
A = SELECT COUNT(*) FROM Student
```

Then, run the query:

```
res = q(CorT1) + q(CorNotT1) - q(A)
```

If res is equal to 1, then return `mid`, which is Natalie's grade.
Otherwise, we proceed with the algorithm of updating variables `low, high, mid`.

For the different tracker `T2`:

```
T2 = SELECT COUNT(*) FROM Student
      WHERE (Name = 'Natalie' AND Score < mid)
```

Set up 3 new queries:

```
CorT2 = SELECT COUNT(*) FROM Student
        WHERE (Name = 'Natalie' AND Score < mid) OR Gender = 'M'
CorNotT2 = SELECT COUNT(*) FROM Student
         WHERE (Name = 'Natalie' AND Score < mid) OR Gender != 'M'
// `A` remains the same
```

Then, run the query:

```
res = q(CorT2) + q(CorNotT2) - q(A)
```

If `res` is equal to 1, then update the value for `low` to be `mid+1`, because
, Natalie's grade is higher than the grade `mid`.
Otherwise, it indicates that Natalie's grade is lower than the grade `mid`, thus update `high` to be `mid-1`.

The whole process described up to this point is `while`-looped with the condition that `low` is less than or equal to `high`.

This Binary-Search algorithm will return Natalie's grade.

(c) No, Table 2 is not 3-anonymous. As stated, if Name, Birthdate, Gender are identifiers, then the statement, “For each released record, there exists 2 or more released records from which record cannot be distinguished.” is violated in Table 2 because the birth dates are all unique.

A 3-anonymous table will be:

Name	Birthdate	Gender	Postal Code
*	Jul-Sept	F	G9Q 3X2
*	Oct-Dec	F	G9Q 3X2
*	Apr-Jun	M	H4A 5A6
*	Jul-Sept	F	Y1R 4J4
*	Apr-Jun	F	H4A 5A6
*	Oct-Dec	F	H4A 5A6
*	Oct-Dec	M	H4A 5A6
*	Oct-Dec	M	H4A 5A6
*	Apr-Jun	M	Y1R 4J4
*	Oct-Dec	M	Y1R 4J4
*	Jul-Sept	F	G9Q 3X2
*	Oct-Dec	F	H4A 5A6
*	Jul-Sept	M	H4A 5A6

Value of l = 2.

Q4. Legal Issues

(a) The petition, raised by Michael Weinberg was to remove the specific language from the exemption, which restricts the use of computer programs in 3D printers for producing commercial goods subject to legal regulations or where circumvention is illegal. The two justifications for it are that firstly, because the language introduces ambiguity, rendering the exemption almost unusable in many instances, and secondly, because the concerns addressed by this clause are more appropriately dealt with by other governmental bodies.

(b) Opponents of jailbreaking voice assistant devices primarily argued that it could lead to piracy issues. They believed that these devices, being less complex than personal computers, had limited security measures and were more susceptible to piracy. They also claimed that jailbreaking could enable the installation of counterfeit and copyright-infringing apps. Additionally, they disputed that jailbreaking was necessary for promoting new app development.
If the exemptions were approved, companies selling such voice assistant devices could revise their terms of service agreements and warranties to explicitly state that jailbreaking voids warranties and support for the device. This would discourage users from attempting to jailbreak their devices, and withdraw support for devices that have went through jailbreaking.

(c) Yes, such circumvention would be possible.
Removing certain provisions argued that the existing language strikes a fair balance between the interests of security researchers, copyright owners, and the public. They believed that the negative impacts claimed by proponents lacked evidence and were based on misinterpretations of the text. Additionally, opponents warned that removing these limitations could lead to excessively broad interpretations, potentially enable copyright infringement, and pose risks to public safety and national security.