Problem 1

Collaboration Statement: My work on this problem is completely independent and I did not collaborate with any students or external sources.

a) There are a relatively small number of brown buildings, and only 1440 minutes in a day (and Bob's space of Yes/No messages is even smaller), so if Sierra has the public keys, she can create a list of all possible messages with their corresponding ciphertexts . Using this, she can look up the ciphertexts, and since the encryption is deterministic, she'll know the corresponding message.

b) At the end of each message, Alice and Bob should add a random (or psuedo-random) 16-digit number. When they receive messages, this value should be treated as junk, but for Sierra, this makes the space of all possible messages much too large to realistically store all of them with their ciphertexts.

c) Sierra is an IND-CPA adversary, since with a polynomially-bounded number of encryptions, she can gain information. She's able to hash any plaintext input she chooses in polynomial time.

Problem 2

Collaboration Statement: My work on this problem is completely independent and I did not collaborate with any students or external sources.

a) Alice and Bob should agree before-hand on a cryptographic hash function to use. Each of them then thinks of an arbitrary integer, which should ideally be done using an n-sided die. On their own, they hash the integer, then send the hash to the other person. Once they've received each others' hashes, they then send the unhashed integer. Each person should hash the number they received, and confirm that it matches the hash. If it doesn't, the other person cheated and they should abort the protocol. If we add the integers and then use the modulo n operation, to get a number from range 0 to $n-1$. This is the randomly generated number.

This prevents Alice and Bob from cheating because they can't feasibly reverse a cryptographic hash function (as it is one-way), so they have no idea what the other person's number is. They also won't send their number until after they've received the other person's hash, so neither of them can cheat. As each person chooses a number uniformly randomly, the outcome is also uniformly random.

b) Yes. For example, Alice and Bob text their hash. Bob then texts his number to Alice. Alice sees Bob's number, and then realizes her initial choice would give her an unfavorable outcome. So, she then cheats and texts an invalid number, ending the protocol and avoiding the outcome.

c) Alice and Bob each text Jake their number. Jake then adds the numbers, performs the modulo operation, and texts both back with the result.

Problem 3

Collaboration Statement: My work on this problem is completely independent and I did not collaborate with any students or external sources.

a) I believe that exceptional access is never justified. While there are legitimate reasons why a government may need access to encrypted data, people should have a right to privacy from their government. Requiring all encryption (or even just some services on a wide scale) is intrusive and opens these systems up to abuse. The GCHQ article mentions that "investigators are skilful in making sure they make the most of every bit (in both senses) of information." While exceptional access would be useful to law- enforcement, it is only one piece of the puzzle in their jobs, and doesn't justify the vulnerabilities that it creates.

b) No, this principle is not met by their proposed excetional access mechanism. The article states that "You end up with everything still being end-to-end encrypted, but there's an extra 'end' on this particular communication." This is a disingenuous interpretation of end-to-end encryption, as people usually take this to mean that the data is safe between transmission from one node to another. Allowing access to this data to a third party would certainly change the trust relationship for some users, since if this ability exists, then it's not guaranteed that it couldn't be exploited by malicious governments or others.

c) I don't find this argument in favor of exceptional access convincing. Given that the government also has use for secret communications (consider national security matters), they generally benefit from reporting vulnerabilities. Also, the article mentions other options for gathering data, such as "covertly entering a suspect's house to copy data." This is less high-tech, but is a valid way of gathering information. Governments and the people benefit from better security systems, and any government hiding security vulnerabilities is acting against the best interests of the people.

Problem 4

Collaboration Statement: My work on this problem is completely independent and I did not collaborate with any students or external sources.

a) We can represent decryption using equations as follows: Base case: $P[0] = IV \bigoplus (C[0] \bigoplus E(key)$ General case: $P[i] = C[i] \bigoplus E(C[i-1])$

b) He can know how much of the start of the message is the same. We can use the decryption equation to show that he knows the bytes at which $P_1[0]$ and $P_2[0]$ are different:

$$P_1[0] \bigoplus P_2[0] = (C_1[0] \bigoplus E(key)) \bigoplus IV \bigoplus IV \bigoplus (E(key) \bigoplus C_2[0])$$

$$= C_1[0] \bigoplus E(key) \bigoplus E(key) \bigoplus C_1[0] = C_1[0] \bigoplus C_2[0]$$

By xor-ing the first part of both ciphertexts, we get the same result as xor-ing the first part of both plaintexts. Similarly, as long as the start of the message is the same, he has similar information about the next parts of the plaintext:

$$P_1[i] \bigoplus P_2[i] = C_1[i] \bigoplus E(C_1[i-1]) \bigoplus E(C_2[i-1]) \bigoplus C_2[i]$$

$$= C_1[i] \bigoplus C_2[i]$$

That is, we would then know what bytes differ between $P_1[i]$ and $P_2[i]$.

c) If $j \in C[i]$, then the bits in $P[i]$ and $P[i+1]$ would be corrupted (should $P[i+1]$ exist).

d) False: each ciphertext depends on previous ciphertexts.

e) True: each plaintext only depends on ciphertexts, not on decrypted plaintexts.

f) This is IND-CPA secure. At best, an attacker can determine where the differences are between two messages, but can't determine which message is which, so the probability of them guessing will be 50%.