Marcantonio Soda
CSE242
HW8

1)

a) When P makes a cryptographic commitment to the randomly chosen value of k, k is bound to that value meaning that P cannot change it later. Importantly, P can commit k without having to share the value with V. Overall, the significance of the cryptographic commitment is that P knows k and V can verify that k has not changed without needing to know the value of k.

b)

$$g^{s} = y^{c} \cdot R$$
$$\Rightarrow g^{xc+k} = y^{c} \cdot R$$
$$\Rightarrow g^{xc} \cdot g^{k} = y^{c} \cdot R$$
$$\Rightarrow g^{xc} \cdot R = y^{c} \cdot R$$
$$\Rightarrow (g^{x})^{c} \cdot R = y^{c} \cdot R$$
$$\Rightarrow y^{c} \cdot R = y^{c} \cdot R$$

P would not realistically be able to produce the exact value of **s** such that g^s=Y^c*R if it did not know **x** because **s** depends on **x** in the same way that **Y** depends on **x**. It is infeasible that P could produce a valid **s** without it having known **x**. This is how V can prove that it knows how to solve discrete logs without having to give the solution to a discrete log.

c)

V only knows c, Y, R, and s and s=xc+k. V does not know **k**. It is not possible for V to figure out **x** without knowing **k**.

2)

Paxos requires the majority of nodes to agree. If the majority does not agree, the paxos algorithm restarts.

3)

Safety: If a customer does not pay, they will not get their food.
Liveliness: If a customer does pay, they must get their food.

4)

a) The fact that asynchronous transmission is not accepted makes it more difficult for a node to be malicious.

b) If the system were asynchronous consensus would be very difficult to reach. It would likely cause forks to happen often because it is imperative for a node to take into account the nodes that came before it, which is much more difficult in an asynchronous system.