In order to answer numbers 5 and 6, I will refer two equations presented in Satoshi's original paper on the bitcoin network.

$$\lambda = z\frac{q}{p}$$

p=the probability that an honest node finds the next block
q=the probability that an attacker finds the next block
z= the number of blocks that have been created since the attack began

Using the equation above, the following equation describes the probability of success of winning the blocks in succession as z goes to infinity.

$$1-\sum_{k=0}^{z}\frac{\lambda^k e^{-\lambda}}{k!}\left(1-(q/p)^{(z-k)}\right)$$

5+6. This equation addresses both 5 and 6; as mentioned in the question, one would merely have to set z=1. The fraction of hashing power controlled by the attacked scales with q, thus s can be used for the probability that the attacker wins the next block.

7. If one wishes to join the pool in the first place, they are looking for more regular and stable profits. My strategy for building a stable mining pool would revolve around punishing those caught cheating rather than trying to prevent it in the first place. Miners caught cheating would be forced out of the pool by the rest of the honest miners. This means they would be trading a one time large profit for the stability of their position in the pool, which is ultimately more valuable in the long run if they joined in the first place. In order to protect against person's looking to join and quickly cheat, there would be a time period in which you must contribute computing power with no profits.

8. I joined up with several classmates to join a mining pool. Due to our lack of technical proficiency on the matter and that most of these people I knew personally, it was generally accepted that everyone was working towards beating the rest of the network as opposed to each other. There were 11 persons in total, only a handful of which had been successful on the Pointcoin network. I trusted these people least, and tried to monitor very carefully that strategies they suggested did not have any loopholes.

9. Our mining pool talked about several attack strategies; the most exciting of these was time jacking, but it ultimately proved too difficult in the technical sense. We then decided that the best way to proceed was to attempt a 51% attack in which we make a transaction, and then erase it by forking the network with a longer chain. This was a very probable solution as we easily had a majority of the computing power in our group.

We began by isolating ourselves from the network. Unfortunately, we did not think out the strategy until it was a little too late: we failed to spend a coin before leaving the network. However, once removed, we then began to compute a competing chain. We were confronted with technical difficulties at this point; 2 unknown nodes were connected to our 11, meaning it was likely that we were still connected to the original network. Assuming that our attack had been foiled by nodes hardcoded into the system, we abandoned the attack. What we did not realize is that these nodes may not have been connected, as we learned the next day that we had indeed forked the network. Had we been able to re-convene the group, we would have changed two parts of the attack: 1, spend a pointcoin before leaving the network and build off the block before it once we leave the network. 2, once we leave the network, use our combined computing power to outpace the network. Since we had the vast majority of the computing power, the more blocks we calculate, the more likely that we beat the original chain. Once our miners began to slow down again, we would rejoin the network, submit the longer chain, and enjoy our double spent coin.