Prob. 1	Prob. 2

Team members: Küng, Pirelli, Schubert, Dousse, Vu

Problem 1.

For all edges (a,b) in a spread set S, either $a \in S \& b \notin S$ or $b \in S \& a \notin S$ or $a \notin S \& b \notin S$. In other words, at most one of (a,b) is in S.

Let's take the complement of a spread set, for each case:

$$a \in S \& b \notin S -> a \notin S \& b \in S$$

 $a \notin S \& b \in S -> a \in S \& b \notin S$
 $a \notin S \& b \notin S -> a \in S \& b \in S$

In all three cases, at least one of (a,b) is in S. This is the definition of a vertex cover.

This means that *SPREAD_SET* can be rephrased in the following way:

Given a graph G of size n and a size k:

- 1) Find a vertex cover for G of size (n-k)
- 2) Take the complement of that vertex cover.

Step 2 is obviously polynomial, therefore *VERTEX_COVER* can be reduced to *SPREAD_SET*. Since *VERTEX_COVER* is NP-complete, *SPREAD_SET* must also be NP-complete

Problem 2.

The shortest path is easy to find in polynomial time using e.g. Dijkstra's algorithm. Checking if this path is shorter than k is also done in polynomial time. So we can say that *SHORT_PATH* is in P.

LONG_PATH is in NP because we can check in polynomial time that it is a path and that its length is greater than or equal to k. If we take a special case of LONG_PATH that is an Hamiltonian Path where k is equal to the number of vertices minus 1, we can conclude that LONG_PATH is NP-complete as we know that finding an Hamiltonian path is NP-complete.