Editorial: Scales

This is based on the well-known problem of ordering five coins with 7 (binary) weighing. There are 120 permutations and 128 possible sequences of seven answers, and it is indeed possible to find a strategy which always works.

In our problem, there are 720 possible answers, and with at most 6 ternary weighing, theoretically we could obtain 3^6 = 729 possible answers. The gap is very small, but it turns out that it is again possible to find a strategy which uses just 6 weighing.

The easiest solution here is to generate a memorization function which, for each possible subset of possible permutations, tries all the possible questions, and returns the best one. It is possible to bound the branching, as follows:

- Always take care that there are at most 243 consistent results after 1 weighing, at most 81 consistent results after 2 weighing, etc.
- If, for the given subset of power P, we have already found a solution which uses N weighing, and P > 3^(N-1), then do not look further (we have already found the best possible answer).

With these optimizations, we could write a program which generates the strategy tree, in the form of a program which could be submitted for judging (see builder.cpp, which does not build it in the right format yet, but is close enough). With the optimizations above the builder runs fast enough, so it is also possible to generate it on the fly.