# CTAP

## Part 1

### i.

The stream cipher implementation, stream.py (written in Python), can be found in appendix A. It can be ran using the Python 2.7.6 compiler with the command *python stream.py*. This will produce the challenge 25 bit output of 3 Lfsrs with initial states 97, 975 and 6420:

1 0 1 0 1 0 1 1 1 0 1 0 1 1 0 0 0 0 0 0 1 0 0 1 1

Include code in appendix

### ii.

<https://www-users.cs.york.ac.uk/~jac/PublishedPapers/AlmostBooleanfunctionsCEC2003.pdf>

* All individual streams have a correlation below 0.6 -> no obvious key for each
* L0 and l2: 228715 (7 bits for l0, 13 for l2)
* Keys: L0: 27, L1: 1357, L2: 7531
* Notes: L0 xor L1 or L0 xor L2 correlate strongly with output (crack pair of keys by brute force quickly)
  + L0 xor L1, followed by L2 (O((2^18)+2^13), will be quicker than L0 xor L2 followed by L1 (O((2^20)+2^11)

## Part 2

1. 45858 (include code in appendix!!)