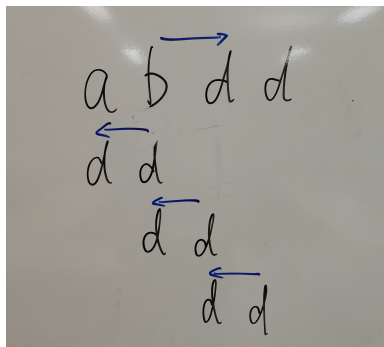# Week 09

Alaukik, Nick, Linfan, Xinyu

4. The idea behind the pattern search is illustrated in the following diagram, where `pattern` is traversed from right to left, and `text` is traversed from left to right.

The implementation is based on the naive search. Possible optimisations include those similar to the Rabin-Karp algorithm and the KMP algorithm: For instance, if `text` has many repetitive characters, we could save the time spent on string comparisons by using a hash function. That is, if the rolling hash is equal to `pattern`'s hash, perform the naive search by traversing `pattern` from right to left. Alternatively, we could also eliminate repetitive matching by exploiting the fact that once a mismatch occurs, the information on where the next match could possibly begin is entirely encoded in `pattern`: