**Lab assignment 1: Exact string matching (Naïve *vs* Rabin Karp)**

1. Given two DNA sequences **seq\_x** and **seq\_y**, find using Naïve approach if one is a subsequence of other. If so, identify the *text* and *pattern* sequences as determined the pattern matching.

|  |  |  |
| --- | --- | --- |
|  | **Test case 1** | **Test case 2** |
| **Input** | ATGCATTGCTGCATTG | TGCATTGATGCATTGC |
| **Output** | text: ATGCATTGCpattern: TGCATTGnumber of matches: 1 | text: ATGCATTGCATTGCATTGCGpattern: TGCATTGnumber of matches: 2 |

1. Solve the above problem using Rabin Karp method.
2. Find the performance improvement in a *Double Hash* Rabin Karp when the substring matching uses 2 hash different hash functions instead of a single function. For the second hash, process the string in the opposite direction to that of the first (*eg.*, AATTGG for GGTTAA).
3. Verify the performance difference improvement obtained through *Improved* Rabin Karp as discussed in a 2014 work titled, *A Novel Pattern Matching Algorithm in Genome Sequence Analysis*, by Ashish Prosad Gope, et al. (IJCSIT).
4. Plot the time complexity as seen through your experiments for the four approaches using the complete genome of *E. coli* ([*Escherichia coli* str. K-12 substr. MG1655](https://www.ebi.ac.uk/ena/browser/view/U00096)). A sample plot for such a study is shown below taken from *A FAST pattern matching algorithm* by S S Sheik *et al.* (doi: 10.1021/ci030463z). Specify the list of patterns and their lengths used for the experiments.

