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	`ATGCATTGC` `TGCATTG`	`TGCATTG` `ATGCATTGC`

Output	`text: ATGCATTGC`
	`pattern: TGCATTG`
	`number of matches: 1`
		`text: ATGCATTGCATTGCATTGCG`
		`pattern: TGCATTG`
		`number of matches: 2`

2. Solve the above problem using Rabin Karp method.

3. 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`).

4. 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).

5. 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). 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.

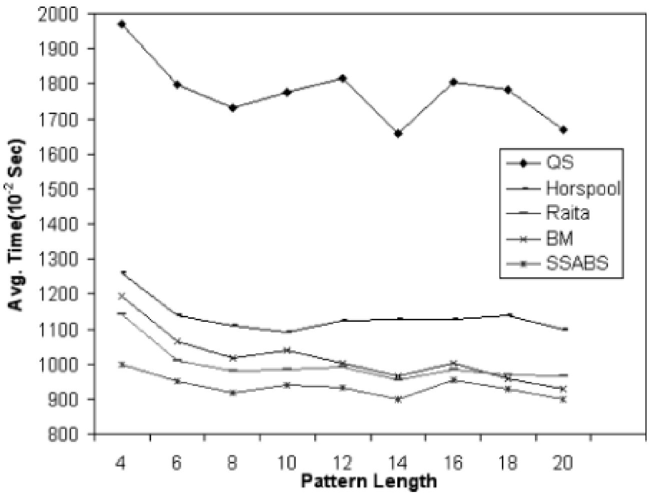

Figure 1. Comparison of the proposed algorithm with the well-known algorithms available in the literature. The database used is the gene sequences comprised of nucleotides ($\sigma = 4$, Table 1). The graph clearly depicts the performance of various algorithms considered in the present study.