

CS 6301 – Implementation of Advanced Data Structures and Algorithms
- by Dr. Balaji Raghavachari

by Bala Chandra Yadav (bxy140430)
Ravindhar Reddy Thallapureddy (rxt140930)
Mohammad Rafi Shaik (mxs146030)

Implementing algorithms to for pattern matching:

The project involves the implementation of various algorithms for pattern matching.

Problem Statement

Compare the running times of the implementation of algorithms for pattern matching.

Software Requirements (Used)

Java version: Java v1.8.0_31

JDK: Oracle Java SE Development Kit 8

Editor: Eclipse

Instructions to run the code (for detailed, please check readme.txt)

JVM arguments:

To compile the file

```
>javac Driver.java
```

To run the program use the following command

```
>java Driver
```

Input(size)		Naïve		rabin Karp		KMP		BoyreMoore	
text	pattern	RT	Memory	RT	Memory	RT	Memory	RT	Memory
1600	9	5	2MB/128MB	1	2MB/128MB	0	2MB/128MB	1	2MB/128MB
5000	9	6	2MB/128MB	2	2MB/128MB	1	2MB/128MB	3	2MB/128MB
20000	19	15	4MB/128MB	10	4MB/128MB	6	4MB/128MB	3	4MB/128MB
50000	35	32	8MB/128MB	15	8MB/128MB	12	8MB/128MB	3	8MB/128MB
100000	51	51	18MB/128MB	20	18MB/128MB	21	18MB/128MB	6	18MB/128MB
500000	153	218	49MB/163MB	53	49MB/163MB	91	49MB/163MB	9	49MB/163MB
1000000	227	394	105MB/230MB	53	105MB/230MB	171	105MB/230MB	13	105MB/230MB
1000000	6000	3790	220MB/554MB	52	220MB/554MB	141	220MB/554MB	18	220MB/554MB

RT is in msec

From the table shown here, we can notice that BoyreMoore outperforms KMP and rabin Karp as input size grows.